DOCUMENT RESUME

ED 388 210 IR 017 359

AUTHOR	Ottmann, Thomas, Ed.; Tomek, Ivan, Ed.
TITLE	Educational Multimedia and Hypermedia, 1994. Proceedings of ED-MEDIA 94--World Conference on Educational Multimedia and Hypermedia (Vancouver, British Columbia, Canada, June 25-30, 1994).
INSTITUTION	Association for the Advancement of Computing in Education, Charlottesville, VA.
REPORT NO	ISBN-1-880094-10-X
PUB DATE	94
NOTE	794p.; For individual papers, see IR 017 360-459.
AVAILABLE FROM	Association for the Advancement of Computing in Education, P.O. Box 2966, Charlottesville, VA 22902.
PUB TYPE	Collected Works - Conference Proceedings (021)
EDRS PRICE	MF05/PC32 Plus Postage.
DESCRIPTORS	*Distance Education; *Educational Media; Educational Technology; Elementary Secondary Education; Foreign Countries; Futures (of Society); Higher Education; *Hypermedia; *Multimedia Instruction; Multimedia Materials; Telecommunications
IDENTIFIERS	Multimedia Technology

ABSTRACT

This collection presents papers pertaining to the wide area of educational multimedia and hypermedia. The conference serves as a forum for the dissemination of information on the research, development, and applications in all areas of multimedia/hypermedia in education across all disciplines and levels of education. The papers cover a wide range of topics including: artificial intelligence; authoring; distance education; navigation; learning by doing; language learning; media in education; pedagogical issues; hypermedia systems; hypermedia applications; small dedicated applications; improving classroom teaching; interactive learning environments; novel applications approaches and ideas; and computer supported cooperative work. Special attention is given to distance education and artificial intelligence in education. The volume is a guide to what is happening in educational multimedia and hypermedia, now and in the future. These proceedings contain 11 keynote and invited papers, 86 full papers, reports of 6 panel discussions, 49 short papers, and reports of 100 demonstrations and posters. (MAS)

Educational Multimedia and Hypermedia, 1994

edited by
Thomas Ottmann
Ivan Tomek

Proceedings of *ED-MEDIA 94*—

World Conference on
Educational Multimedia and Hypermedia

Vancouver, BC, CANADA; June 25-30, 1994

AACE Association for the Advancement of Computing in Education

Educational Multimedia and Hypermedia, 1994

edited by
Thomas Ottmann
Ivan Tomek

Proceedings of *ED-MEDIA 94*—
World Conference on
Educational Multimedia and Hypermedia
Vancouver, BC, CANADA; June 25-30, 1994

AACE Association for the Advancement of Computing in Education

Published by

Association for the Advancement of Computing in Education (AACE)
P.O. Box 2966
Charlottesville, VA 22902 USA

Printed in the USA

ISBN 1-880094-10-X

PREFACE

ED-MEDIA 94 - World Conference on Educational Multimedia and Hypermedia is the second international conference on this subject sponsored by the Association for the Advancement of Computing in Education (AACE). It is also a continuation in the series of International Conferences on Computer Assisted Learning (ICCAL), held for the first time in 1987 in Calgary and then held three more times in Dallas, Hagen (Germany), and Wolfville (Canada). However, ED-MEDIA is not only a continuation of ICCAL and broadens its scope considerably by including all aspects of educational multimedia.

The first ED-MEDIA conference held in 1993 in Orlando, Florida, served as an excellent multi-disciplinary forum for the dissemination of information on the research, development, and applications in all areas related to multimedia/hypermedia in education across all discip'ines and levels of education. By again enlarging the scope and by adding weight to distance education and artificial intelligence in education, ED-MEDIA 94 is even more attractive and is establishing the series as a *must* for everyone who wants to obtain an overview of the field and its most recent developments. This is especially so since although the essence of ED-MEDIA is a collection of carefully selected submitted papers, ED-MEDIA 94 also includes thirteen tutorials and workshops, six keynote speeches by leading industrial and academic personalities, eight invited lectures, panel discussions, and an exhibit.

This proceedings volume contains the papers selected by the Program Committee for the conference and the invited lectures presented by distinguished scholars in the field. The papers cover a wide range of topics including artificial intelligence, authoring, distance education, navigation, learning by doing, language learning, media in education, pedagogical issues, hypermedia systems, hypermedia applications, small dedicated applications, improving classroom teaching, interactive learning environments, novel applications approaches and ideas, and computer supported cooperative work.

ED-MEDIA 94 occurs at a time when public and research attention focus on the concept of information highways, the integration of computer and TV technology, the advent of worldwide hypermedia networks, CD-ROM technology, and many more exciting technological developments that will have considerable impact on the future of learning and teaching. One of the roles of the conference is thus to provide a glimpse into the possible forms of this future both in terms of work that is now in the labs and the visions of the leading workers in these fields. The proceedings in your hands summarize the results presented at the conference. As is always the case, however, proceedings can reflect a conference concerned with the role and potential of educational multimedia and hypermedia only imperfectly. We hope, however, that reading of the written form of the presentations and examination of the program will provide a picture of the broad spectrum of topics covered in Vancouver and stimulate you to attend next year's conference in Graz, Austria, personally.

The co-chairs of the Program Committee express their thanks to all those who contributed to the success of this conference. In the first place, we thank the keynote and invited speakers, authors of presented papers, developers who demonstrated their results, exhibitors, and panelists. In this context, we also thank the authors of papers that could not be accepted this time because of the very large number of submissions. Finally, the members of the Program Committee have undertaken the difficult task of refereeing the more than 400 papers submitted for the conference, and we are grateful to them for the time that they put into reading the submissions and evaluating them.

Special thanks are owed to AACE and its staff, in particular to Dr. Gary Marks, as well as Dr. Hermann Maurer, the creative force behind ICCAL and ED-MEDIA conferences. Without their help, ED-MEDIA 94 would not have been possible.

Prof. Dr. Th. Ottmann
Universitaet Freiburg, Institut fuer Informatik
Rheinstr. 10-12, D-79104 Freiburg, Germany
Tel.: ++49 +761 203 5687, Fax: ++49 +761 203 5686
E-mail: ottmann@informatik.uni-freiburg.de

Dr. Ivan Tomek
Jodrey School of Computer Science
Acadia University, Wolfville, Canada B0P 1X0
E-mail: ivan.tomek@acadiau.ca

ED-MEDIA Conference Steering Committee

Chair: **Hermann Maurer**, *Univ. of Auckland, New Zealand*
David Jonassen, *Pennsylvania State Univ., USA*
Gary Marks, *AACE, USA*
Douglas Norrie, *Univ. of Calgary, Canada*
Thomas Ottmann, ***Prog. Com. Co-Chair***, *Univ. of Freiburg, Germany*

ED-MEDIA 94 Program Committee

Co-Chair: **Ivan Tomek**, *Acadia Univ., Canada*
Demos/Posters Chair: **Pat A. Carlson**, *Armstrong Lab, Brooks AFB, USA*
Panels Chair: **Marc H. Brown**, *DEC Systems Res. Center, USA*
Workshop Chair: **Thomas Reeves**, *Univ. of Georgia, USA*
Philip Barker, *Teesside Polytechnic, UK*
Ian D. Benest, *Univ. of York, UK*
Peter Brusilovsky, *Int'l Center of Sci. & Tech. Info., Russia*
Gordon Davies, *Open Univ., UK*
Alain Derycke, *Univ. de Lille, France*
Jan Gecsei, *Univ. de Montreal, Canada*
Peter Gloor, *Dartmouth College, USA*
Ricki Goldman-Segall, *Univ. of British Columbia, Canada*
Thomas W. Greaves, *IBM, USA*
Greg P. Kearsley, *Geo. Washington Univ., USA*
Nadia Magnenat-Thalmann, *Univ. of Geneva, Switzerland*
Fillia Makedon, *Dartmouth College, USA*
Gary Marchionini, *Univ. of Maryland, USA*
Cliff McKnight, *HUSAT Research Institute, UK*
Maria Teresa Molfino, *Consiglio Nazionale delle Ricerche, Italy*
Tony Montgomery, *Royal Melbourne Inst. of Tech., Australia*
Max Mühlhäuser, *Univ. of Karlsruhe, Germany*
Graham Oberem, *Univ. of Washington, USA*
Setsuko Otsuki, *Kyushu Institute of Technology, Japan*
Nick Sherbakov, *Graz Univ. of Technology, Austria*
Gerhard Strube, *Univ. of Freiburg, Germany*
Martial Vivet, *Univ. du Maine, France*
Beverly Woolf, *Univ. of Massachusetts, USA*
Yoneo Yano, *Tokushima Univ., Japan*

❖

Table of Contents

Keynote and Invited Papers

Full Papers

PANELS

SHORT PAPERS

DEMONSTRATIONS/POSTERS

Keynote and Invited Papers

Educational Multi-Media in a Networked Society

A.W.BATES
Research and Strategic Planning
The Open Learning Agency, 4355 Mathissi Place, Burnaby, British Columbia V5G 4S8 Canada
E-mail: tonyb@ola.bc.ca

Abstract: Multi-media is developing fast due to the convergence of different media (text, data, graphics, video, audio) into a common digital platform. However, there is another convergence that is also important: the convergence of telecommunications, television and computing. Parallel to these technical convergences is another set of developments in work and leisure that is opening up new markets for education and training, and which requires the creation of new institutional models. This paper explores the instructional and organizational implications of these developments.

Note: this paper is the summary of a multi-media presentation; most of the detail supporting this paper is in the presentation.

The technical and economic revolution

Multi-media in education has been seen by many primarily as an extension of computer-based learning. This is understandable, as some of the main constraints on computer-based learning have been the high cost of incorporating good graphics and video materials, and the restricted sensory stimulation for learners from screen-based text. The addition of high quality graphics, audio, and video to text, and more powerful editing and authoring software, provide a major enhancement of computer-based learning. The costs of hardware and the cost of producing multi-media materials are also dropping rapidly. 'Stand-alone' computer-based learning will become even more powerful as artificial intelligence and virtual reality develop. However, while 'stand-alone' applications of multi-media will continue to be important in education, a much more significant development will be the application of high-speed multi-media networks for educational purposes.

As well as the convergence of different media within a common computer platform, we are also seeing the convergence of the previously separate technologies and industries of computing, telecommunications and television. For instance, in April of this year, Stentor, an alliance of Canadian telephone companies, announced an $8 billion, 10 year initiative, called BEACON, that will bring broadband, multi-media services to 80%-90% of all homes and businesses in Canada by the year 2004. The social and educational impact of this convergence, and the speed with which it will be implemented, will be revolutionary and deeply challenging to established educational institutions.

At the same time as this technological revolution (and partly because of it), the needs of the workforce are also rapidly changing. In 1993, 78% of all jobs in the USA were in service industries, and the trend is likely to continue (Economist, 1994). Microsoft's annual revenues are greater than Sony's and Honda's combined, but they employ 100 times fewer workers. Most new jobs are being created in Canada by companies with less than 20 workers; indeed, the trend to both self-employment and working from home is likely to grow (StatsCan, 1992).

The wealth of nations will depend increasingly on knowlege-based, high-tech industries, in the areas of bio-technology, environmental products and services, computer software, financial services, and entertainment (particularly film and television). Furthermore, these are highly competitive, global industries. Keeping even a few months ahead of the competition, in terms of innovation and knowledge, are critical to survival, as is the quality of product and service. This means that education and training, not just in the pre-work years, but throughout a lifetime, are essential elements of a successful work-force. However, if every worker currently in the workforce was sent back to college for three months training every five years, we would have to double the post-secondary education system in Canada. In practice, of course, the political trend is to reduce or limit public expenditure, to make schools and colleges more cost-effective, to take greater numbers for less cost.

Multi-media and modern telecommunications do offer an opportunity to meet these lifelong learning needs of the work-force in a cost-effective manner. This will not happen though without thinking very differently about how education and training will be organized, in order to serve the needs of the work-force.

Learning in the 21st century

Modern learning theory sees learning as an individual quest for meaning and relevance. Once learning moves beyond the recall of facts, principles or correct procedures, and into the area of creativity, problem-solving, analysis, or evaluation (the very skills needed in the work-place in a knowledge-based economy - see Conference Board of Canada, 1991), learners need inter-personal communication, the opportunity to question, challenge and discuss. Learning is as much a social as an individual activity. However, for someone working in a small company, the nearest person with similar interests and expertise may be somewhere on the other side of the country, particularly in leading-edge technologies, and particularly in Canada.

Work and learning will be inseparable. Most learning will be informal and lifelong. It is not difficult to build a convincing portrait of learning at the work-place. We can envisage a computer software designer or television animation artist, called Sue, probably working from home, needing information on a certain technique or approach, or advice on how best to create a certain effect. From previous experience and contacts, or on the advice of a colleague, she has the name of someone half-way across the country (Wayne). From her work-station, Sue calls Wayne, talks about the problem, and Wayne loads up some software which he 'shares' with Sue via the network. Sue asks a few questions, tries a couple of things on-line while Wayne watches and comments, then downloads the software. Sue and Wayne are both registered with an educational institution that has been set up to enable the exchange of commercially sensitive material for learning purposes. Sue's work-station has automatically displayed the cost per minute of consulting Wayne, and the cost of rights for downloading the software. However, Sue was also able to give Wayne some information, and this is charged back to Wayne's account. Sue now not only has the software she needs, but also can contact Wayne (on a chargeable basis) any time she has a problem with the software. The learning context has been established. Note it is fragmented, on demand, and charged at cost.

Learners will interact with their desk-top or portable workstations in a variety of ways, determined by the nature of the learning task, and their preferred style of learning in the work situation. These preferred styles will vary considerably, both within a single person, depending on the task, and, for the same task, between different individuals.

The learning context will need to encompass the following:

- working alone, interacting with learning material (which may be available locally or remotely);
- working collaboratively (and in an equal relationship) with fellow workers at different remote sites, either synchronously or asynchronously: both these modes are likely to be multi-media;
- as an 'apprentice' or 'student', working with a more experienced worker, supervisor, or instructor;
- as an instructor, supervisor or more experienced colleague for other less experienced colleagues.

The same person may find themselves in each of these roles within a single working day. Learners will also need to be able to work from home, or from a work-site, or while in transit. They will need the following:

- access to information (searching, downloading) from multiple sources in multiple formats
- selection, storage and re-ordering/re-creation of information
- direct communication with instructors, colleagues, and other learners
- incorporation of accessed/re-worked material into work documents
- sharing and manipulation of information/documents/ projects with others.

Learners will need to access, combine, create and transmit audio, video, text, and data as necessary.

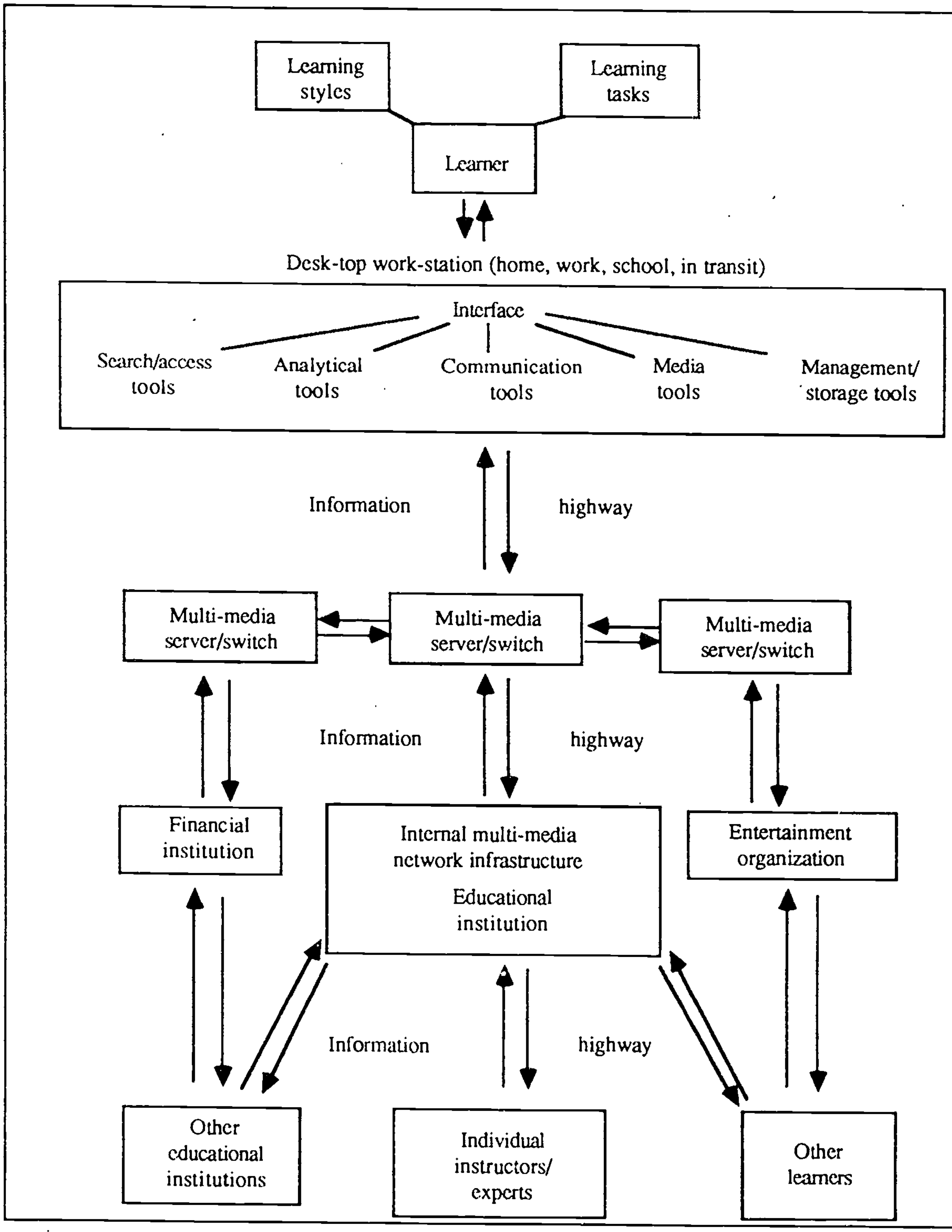


Figure 1: A networked educational multi-media system

If we take this as the design requirement, there is then a need to build *systems* that support this form of learning, both for formal and informal learning. I give my own personal 'vision' of how such a system would provide the kind of educational experiences I would like to see. This is summarized in Figure 1, above.

The work-station

Arguments about whether access to the information highway will be through a personal computer or a 'converter' on a television set miss the point. The work-station of the future will be a multi-purpose machine, probably in modular form, including input (voice, pen, keyboard, gestures) and display (screen, sound, printer) devices, telecommunications, computing and television. It will be at least in part portable.

Key features will be the interface between the user, the tools available to the learner within the workstation, and a range of remote services, both educational and non-educational, that can be accessed remotely via the work-station.

The interface

Design work has already begun on building interfaces for the information highway. The Virtual Interactive Environment for Workgroups (VIEW) is one such system currently in the initial stages of development in Canada by MPR Teltech, the Open Learning Agency, Simon Fraser University, Science World (British Columbia), the British Columbia Educational Technology Centre, and Stentor. The VIEW system will provide tools for creating and using 'multimedia conferences', and for enabling users to engage in individual and collaborative group activities using information from diverse sources and in a variety of media formats, operating either in synchronous or asynchronous modes (Teles and Laks, 1993).

In essence, when learners switch on their work-station, there is a window with a choice of services. One of the choices (others may be films, home shopping, financial services, messages, etc.) will be education and training. When the learner chooses education and training, VIEW will provide a new window, with a choice of educational services, and a choice of a range of software tools to facilitate the learning and communication process. Thus learners will be able to search, access and download information from a variety of sources in a variety of media formats. An example of how this will work is given in the presentation.

The tools

A critical element are the software tools available in the work-station. As well as tools for communication, management and storage of information, there will also be tools that assist in searching, accessing and compressing information, in analysing accessed data for relevance, in 'grouping' appropriate types of information, and tools for importing different types of media-based information, editing, and exporting them. These tools will need to be intuitively simple to use.

The educational institution

What will make or break such a system will be the creation of new organizational structures for educational institutions to provide the administrative and educational support for lifelong learners.

Roles for 'electronic' educational institutions

The critical roles of an 'electronic' educational insitution built to meet the learning needs of the 21st century will be as follows:

- to provide information on education and training needs and opportunities
- to provide quality control
- to provide accreditation, through independent assessment of learning
- to develop coherent curricula, where appropriate

- to provide the service that will make the use of communications to import and export multi-media learning materials easy and user-friendly
- to network learners and instructors
- to create high quality educational multi-media materials in an easily accessible form
- to conduct research into education and training needs
- to apply new technologies, as they develop, to education and training, and to evaluate their use.

Note though that many of the instructors or tutors that are used will not 'belong' or work for the educational institution; many will be independent contractors, or working full-time in a knowledge-based industry, or working for another educational institution. Nor will learners necessarily be 'registered' with that institution, in the sense of taking all or any courses. The institution is primarily a facilitator of learning. In the example of Sue and Wayne, all the educational institution may do is bill, and collect and deliver payment, regarding fees and royalties, to and from Sue and Wayne, the owners of the software, and possibly the telecoms companies (plus a service charge). In other cases, it may offer a full program to groups of students with its own instructors and multi-media materials, leading to its own credential. In others, it will be like a multi-media reference library, with learners just accessing the information they need. It will be quite a commercial organization, collecting fees for many of its services, where this is appropriate.

The internal multi-media network infrastructure

The heart of this service is the internal multi-media network infrastructure, that allows the institution to access, create and deliver educational multi-media services in a variety of formats and a variety of modes. The Open Learning Agency is developing an integrated information management approach that will include both administrative and instructional systems. Basically, learning materials can be accessed, created and stored digitally in any format (video, audio, text, graphics, or any combination). Course designers can access this material electronically, re-edit and re-create learning materials, store and export this learning material in a variety of ways (print, CD-ROM, or down-loaded to local work-stations), depending on the learners' needs. The system will allow for the tracking of materials and services, the on-line payment of fees and charges for services, and student or client record-keeping (including grades and credentials), as well as providing management information on finances and learner activities. This infrastructure is connected through the information highway to multi-media servers or switches. More details of the Agency's plans for its internal multi-media network infrastructure are given in the presentation.

Implications for learning

While schools, colleges and universities will still have reason to provide campus-based learning to groups of learners over set terms or semesters, for social and for some instructional reasons, a great deal of learning will take place outside of this context. Full-time students will in any case soon be a minority in Canadian universities and colleges (63% of all college enrolments in British Columbia were part-time in 1992/93 - B.C. Ministry of Skills, Labour and Training, 1993). Even for full-time students, it will be difficult to categorize them as either 'campus-based' or 'distance education' students within a few years. They will be accessing information and communicating with their instructors, other students, and other subject experts outside their own institution, through multi-media telecommunications, from home and the work-place. Furthermore, multi-media telecommunications will allow them to do this whenever they want, in small chunks as well as in whole courses or programs of study, thus making learning more flexible and accessible, to all ages of learners, and not just young people in the formal system. Learners will also have a much wider choice of sources of learning, being able to access expertise from anywhere in the world.

As important as the context of learning will be the approaches to learning and instruction. Multi-media telecommunications will allow learners and subject experts to engage in dialogue, questioning, and exploration of a wide range of alternative approaches, as well as the sharing and joint working of multi-media documents. Programs and instruction can be tailored to the needs of each individual. Multi-media telecommunications can also encourage collaborative approaches to learning. Learning will often occur without the direct intervention of a 'formal' instructor, through the use of peers and people working in a job but who have expertise. Most important of all, as they learn through multi-media telecommunications, people will use the same tools and develop the same skills that will be an essential part of their work and leisure activities. 'Stand-alone' multi-media applications will still have an important role to play, especially where learners need to work through carefully a disciplined set of principles and ideas, or need a great deal of practice and experiment to fully understand a subject of study. Their use will increase if learners are given the opportunity to re-work and re-create their own multi-media applications, as projects or for the purposes of assessment. However, stand-alone applications will be a specialized and relatively limited use of multi-media within a much richer learning environment, that will include two-way communication and the transporting of multi-media materials between learners, and between the learner and a mentor. (More details of the different curricula approaches made possible through multi-media telecommunications are given in the presentation - see also Bates, 1993).

Conclusion

This vision for a system is not a utopia, nor even many years away. The wideband highways are at this moment being constructed, and should be in place within 10 years. Multi-media switches, using ATM technology, are now being built. Interfaces to the information highway, and software tools to facilitate multi-media learning, are being designed. The software for handling multi-media communications is being developed by companies such as Oracle.

However, the most difficult part of the system to put in place will be an appropriate educational infrastructure to support the kind of learning needed in the 21st century. The provision of appropriate education and training services to run on the information highway is critical; there is no automatic guarantee that people will use the information highway to an extent that justifies the cost of investment, if services are not provided that meet people's needs. Unfortunately, existing educational institutions were created to meet the needs of a society that are fast disappearing. We need new educational organizations that can exploit the information highway to meet the needs of the 21st century. Economic development will depend as much on the success of creating and supporting such organizations, as on establishing the technological infrastructure. It is critical to get this right because there is no doubt that those countries that harness the power of multi-media communications for education and training purposes will be the economic powerhouses of the 21st century.

References

Bates, A.W. (1993) 'Educational aspects of the telecommunications revolution' in Davies, G. and Samways, B. *Teleteaching* London/New York/Amsterdam: North Holland

B.C. Ministry of Skills, Labour and Training (1993) *Environmental Scan* Victoria, B.C.: Ministry of Skills, Labour and Training, Policy, Planning and Program Evaluation, p.28

Conference Board of Canada (1991) *Employability Skills Profile: The Critical Skills Required of the Canadian Workforce* Ottawa, Ont.: The Conference Board of Canada

Economist (1994) 'Schools brief: the manufacturing myth' *Economist* Vol.330, No.7855 (March 19-25, 1994), p.91

StatsCan (1992) *Labour Force Survey* Ottawa: Statistics Canada

Teles, L. and Laks, A. (1983) *Virtual Interactive Environment for Workgroups: A Broadband Educational Application* Burnaby, B.C.: The Open Learning Agency

Is There Computer Graphics After Multimedia?

KELLOGG S. BOOTH
Department of Computer Science
The University of British Columbia, Vancouver, British Columbia, V6T 1Z4, Canada

Abstract: Computer graphics has been driven by the desire to generate real-time imagery subject to constraints imposed by the human visual system. Modern workstations employ special-purpose hardware and software in a "graphics pipeline" to satisfy these constraints. We expect that the next generation of hardware and software may meet the requirements of the human visual system and that subsequent generations will vastly exceed them. Multimedia faces the same challenges, but only for pre-existing imagery because image generation takes place outside the multimedia workstation. It will be interesting to see what becomes of computer graphics when off-the-shelf systems have full multimedia capability and when standard computing engines render imagery faster than real-time. A dedicated pipeline for graphics will be redundant for all but the most demanding applications; imagery available today only on expensive systems will be supported by standard components. This talk will review the technical obstacles to achieving this vision and the impact that we can expect to see in educational uses of computer graphics after the research challenges we face today are met.

Many of the revolutions that have swept the world of computing have promised to usher in a new era in education. None have. There's a reason for this, but it's one educators don't like. The reason is that education is not an end in itself and thus education never leads in the adoption of new technology. Computers are in schools now because the world (and this means the business world) uses computers, so students have to be trained in the use of computers in order to function in the business world. Despite many noble efforts by the education community, realistically it is hard to point to any significant impact of computers in education other than small, isolated successes that are often the result of substantial investments of time, money and good will on the part of educators and the computing community.

A particularly good example of this phenomenon is computer graphics. Who could doubt the potential for enhancing a curriculum by using interactive graphics to not only bring a possibly dull subject to life but to add 3-D realism and maybe even a glimpse of virtual reality thrown in for good measure? But it rarely happens. The reasons, as we all know, are two-fold.

The first reason that computer graphics has not had a major impact in education is that despite the many advances that have been made in the field of computer graphics, it remains a fact that very few professionals in any field have access to 3-D graphics workstations and, while a great many people use 2-D computer graphics, the computers generally available to schools do not support the functionality necessary to run many of the applications representative of the state-of-the-art in computer graphics. With time, this may change. Costs are coming down, so as computer graphics comes to have a greater impact on everyday life we might expect, just as typewriters found there way into schools when typing became a part of everyday life in the business world, so too will computer graphics be part of the curriculum once it is adopted in the workplace. (Eventually even electric typewriters made it into the classroom.)

The second reason that computer graphics has not had a major impact in education is less likely to be resolved so easily. This reason is related to the difficulties associated with developing and maintaining courseware. To continue the analogy with typewriters, we can observe that by putting a typewriter into a classroom students are instantly enabled to use the typewriter with perhaps only the need of an introductory text on typing skills. No other expenditure is required. An educator trained in typing can probably do a credible job of bringing beginners to an intermediate level of typing skill. There is an immediate payoff. Students are then able to type the papers they write for school (helping both them and their teachers) and they gain a practical skill for their future life in the world of business.

Not so with computers. Courseware is often quite different from the programs used in business (unlike typewriters, which are used identically in schools and in business). And courseware costs a lot to develop, test and maintain. Courseware that utilizes computer graphics is often much harder to write than normal courseware. Ironically, the advent of GUIs (graphical user interfaces), which were invented to make computers more accessible to non-programmers, make the programming of user interfaces much more difficult than it ever was before and thus, to a large extent, make good courseware even more difficult to produce.

Multimedia is the latest darling of the computing community that many predict will change the way that education is done. Why should we believe this will happen any more for multimedia than it did for computer graphics or for the many other areas of computer science that did not, in the end, have an impact on education?

The remainder of this talk will address this question by first discussing the relationship between computer graphics and multimedia and then by pointing out the differences that are likely to be significant factors leading to the ultimate success of multimedia as an educational tool where computer graphics and others have demonstrably failed. Chief among these is the distinction that will be drawn between multimedia as a means for communication and computer graphics as a means for computation.

A (Very) Brief History of Computer Graphics

For three decades research in computer graphics has had as one of its main concerns the development of hardware and software to accelerate the so-called "graphics pipeline" that supports interactive real-time display of computer-generated imagery (Foley, et al., 1990, section 18.3). Many architectural advances were made to facilitate high-performance 2-D and 3-D graphics applications in specialized domains (Akeley & Jermoluk, 1988; Haeberli & Akeley, 1990). About a decade ago this effort merged with mainstream computing when 2-D and 3-D graphics workstations became the platforms of choice even for applications that often were not primarily graphical in nature but which took advantage of features common to graphics workstations to enhance productivity (IEEE, 1985; IEEE, 1988). Today most components of the graphics pipeline are available as standard features or modest upgrades in common computing environments and the distinctions between personal computers, graphics workstations and traditional mainframe computers have become blurred.

Many of the performance limitations that currently exist for high-quality graphics workstations are related to human perception (screen resolution, color resolution, refresh rate, and update rate). These are about to become non-issues during the next decade when the multimedia revolution eliminates most of the remaining distinctions between interactive real-time graphics and "normal" computing. Every computer will be accessed as a multimedia workstation because almost all user interfaces will employ some aspect of multimedia. When this happens, computer graphics research will become a very specialized area concerned with rather arcane aspects of modeling and rendering whose solutions will be achieved primarily with cleverer, more accurate algorithms that require more computing power, rather than with new architectural breakthroughs.

The graphics pipeline will be replaced with a ubiquitous "multimedia pipeline" that will eliminate the distinctions between real-time and pre-computed imagery, between local and remote imagery, and between real and synthetic imagery.

The Perceptual Bottleneck

We are small factors of two away from meeting most of our goals for computer graphics displays if we simply look at the level of performance currently available and compare it to the strictest requirements we expect to encounter in typical applications. For the most part, these requirements are based on properties of the human visual system that are studied in psychology and related fields. Many of the limitations are discussed in detail in a recent survey article on virtual reality (Ellis, 1991). We will quickly review the ones that relate to multimedia.

Deering has commented on the deficiencies in spatial resolution for current head-mounted displays by pointing out that if this were our regular vision, we would be legally blind (Deering, 1992). But for normal display monitors, the situation is much better. Screen resolution ranges from roughly 512 × 512 for anyone who cares to invest in a graphics board for a PC to almost 2K × 2K resolution for high performance workstations. One sees predictions of 4K × 4K and even 8K × 8K, but it seems that most applications will not be limited by the spatial resolution of a display once another one or two factors of two are achieved in the number of pixels per inch on a screen. Dreams of larger screens will always be with us, but there is a practical limit to the physical size of the screens we expect to use on an everyday basis.

Chromatic resolution is, roughly speaking, the number of bits per pixel used to represent color information. On low-end machines, 8 bits per pixel is still not standard, but this is changing. Even a few years ago it was rare on a PC but it is now becoming more common. Frame buffers have had 24 bits per pixel (8 each for red, green and blue) for many years with 10 and even 12 bits per pixel not uncommon. Certainly 64 bits per pixel (16 each for red, green, blue and an alpha or opacity channel) is adequate given the properties of the human visual system (Levinthal & Porter, 1985), and 128 bits (32 bits per channel) seems overly generous. So again we are within one or two factors of two from the ultimate chromatic resolution we might imagine as being necessary.

Temporal resolution comes in two forms: refresh rate, the number of times per second that a CRT screen is painted, and update rate, the number of times per second that the image being painted is modified (Baecker, 1979). A rule of thumb is that refresh must be above 40 times per second (television is either 60 or 50, depending on whether you live in North America or not, and film is 48 except for a few special projection systems). Lighting conditions and other environmental factors may increase the required update rate to perhaps 120 times per second, but even assuming 240 times per second (which might be necessary for stereo displays), the "one or two factors of two" assumption applies here as well.

In the flight simulator community, the required update rate is generally assumed to be 10-20 times per second (television is 30, film is 24, and Saturday morning cartoons are 15 or less). In virtual reality, head-coupled displays may have somewhat stricter requirements (Deering, 1992; Ellis, 1991). But at worst the update rate needs to be the same as the refresh rate, but the two are solving different problems (refresh rate governs persistence of vision, whereas update rate provides an illusion of continuous motion) and hence there is a distinction.

The fact that all of the performance figures for actual displays are not too different from the upper bounds that are cited above is not coincidence. Most of the current numbers exist because they are a good compromise between image realism and economic realism. So which ones are likely to change most rapidly?

HDTV is about to double or triple the spatial resolution of consumer video products. This will no doubt impact workstation displays, at least by reducing costs (due to economies of scale) and perhaps by increasing resolution as well.

Chromatic resolution is really an issue of more bits of storage (and transmission bandwidth) per pixel plus the costs of digital-to-analog converters (DACs). Consumer televisions are not yet digital. So they don't have DACs. When they are digital, we can expect to see prices drop a lot for DACs and no doubt we will see higher resolution ones at affordable prices. Today, 8-bit DACs are standard computer components and 10-bit and 12-bit DACs are used for high-performance systems.

Refresh rate is likely to be tied strongly to consumer video. So the one or two factors of two may be quite a while in coming. But if the often-predicted shift away from CRTs finally does take place, whatever replaces CRTs (liquid crystal displays?) may provide higher refresh rates and more spatial resolution too.

Update rate is mostly a question of memory, transmission bandwidth and computing power. It will increase almost automatically with the general advance of computing technology.

All of these powers of two compound to maybe a factor of 50 to 100 increase in performance. This is a lot, but it is small compared to the changes in the computer industry achieved over a typical decade. Moreover, once the limitations are overcome, they are likely to be entirely over-run (at least those achieved through raw computing power) because the technological advances are being driven by other forces that will not be satisfied.

Something that is important for multimedia, where images are generated separately from when they are displayed, but not for computer graphics, is compression. Computer graphics is often not networked because the real-time requirements for image generation are more easily met when there is a tight coupling, through the graphics pipeline, of the data structures representing the underlying model and the final representation of the image as a raster array of pixels.

Some early line drawing systems did explore the separation of modeling and rendering across a local network, with transmission bandwidth being the limiting factor that determined the distribution of function between the modeler and the renderer (van Dam, Stabler, & Harrington, 1974). The "wheel of re-incarnation" is a phenomenon in computer graphics that happens because we need to generate the images close to where we view them and we need to use special purpose hardware (Myer & Sutherland, 1968): as display processors increase in complexity there is a tendency to split the work and distribute it across some communication channel between a main processor and a satellite processor, then the satellite processor is made more powerful to gain performance until it is itself a critical resource, and so another processor is spawned off (thus completing another circuit around the wheel). Until recently, network transmission bandwidth was inadequate for this division of labor with raster images. Higher-speed networks and better image compression algorithms are starting to tip the balance again in favor of distributed systems.

Where does this leave computer graphics? With many of the hardware problems close to being solved and the promise of widespread multimedia applications likely to bring costs down, will there be any difference between computer graphics and multimedia? I think so. Moreover, I think the difference is significant and one that will in the end relegate computer graphics to the back seat.

The Multimedia Pipeline

The much touted Information Superhighway is a mechanism for transmitting multimedia data via universally accessible networks. There are few applications that will require the bandwidth being planned for unless they are using accessing multimedia documents. So we can expect to see a situation that already exists for many high-end multimedia users. Typical workstations will both send and receive highly compressed imagery over a network utilizing specialized components designed to support multimedia applications.

If the workstation is generating imagery, there will need to be a data path that starts where the traditional graphics pipeline started (with a data structure describing a scene to be rendered, usually a dynamic scene whose content and viewing parameters change with time) and ending up on the network feeding the compressed representation of the rendered scene to other workstations or archival storage servers. Similarly there will need to be a data path that accepts compressed imagery from the network and puts it onto the screen for when the workstation is displaying imagery.

Why go to the trouble of compressing, transmitting, and decompressing the image? Why not send the description of the scene and let the remote workstation generate the image? After all, we argued above that future workstations will be able to generate imagery faster than real-time for most applications. The answer is that the scene description will be bigger than the image! We noted above that current image sizes of $1K \times 1K \times 24$ bits are within striking distance of what we may need in the future. Yet scene descriptions grow without bound and, much worse, they are idiosyncratic in terms of the primitives they support and the auxiliary information they require (textures, bi-directional reflection functions, etc.). Standardization is hopeless at this time and for the foreseeable future. But raster images, even with arcane compression schemes, are easily described and, at least in principle, easily translated into various formats.

Example: PostScript is the standard way to ship text and images, regardless of how they were produced. Multimedia formats can be expected to play this same role, hiding many of the details behind how an image was produced. A related point is that PostScript delays the final binding of an image to the physical device on which it will be displayed (or printed), but much of the "value" of the program that prepared the image is added before it is translated into PostScript. Future multimedia servers may in fact be selling not the images (or streams of images) they produce, but the technology that produced them, much like Hollywood sells the movies that go on to video casettes for home viewing.

What will we do with all of the multimedia images that arrive over the network? We'll paste them into windows. Which means we will need to integrate hardware and software support for common windowing operations into the multimedia pipeline. Compositing functions will be common place; every workstation will have the equivalent of a video special effects box.

So why will we need to have a graphics pipeline any more? Just compute the images ahead of time (like maybe 1/240 of a second before you need them), compress them using high-speed circuitry designed for the task, move them to secondary memory (to save precious main memory), then fetch them all back when they are needed (1/240 of a second later), decompress them, and finally paste them onto the screen blended with whatever else is being looked at. Sound wasteful? Sure. But the hardware will be (almost) free and otherwise idle, so why not use it. And it will be cheaper than specialized graphics hardware because that will have a very limited market compared to multimedia hardware.

So we again ask, where is computer graphics in all of this? There will certainly remain problems at the frontier of computer graphics. There will always be imagery that is difficult to produce and research will be needed to determine the best techniques. But this will be of little concern to the average user. A second comparison will illustrate this point.

Example: Numerical analysis is not concerned at all with the design of floating point processors, just the standards employed in the representation and the conventions for exception handling and rounding. Instead numerical analysis concentrates on the fundamental algorithms. It has largely returned to its roots in mathematics, leaving the hardware details to computer architects whose concern is mostly speed and not accuracy. Computer graphics will be the same, concentrating on algorithmic aspects of imagery and ignoring many of the hardware details that have previously dominated many of the discussions of new techniques.

Whither Computer Graphics?

Hermann Maurer has argued convincingly that multimedia fills a void resulting from what he calls "the missing organ" (Maurer, 1992). Humans have a highly developed vision system, but little or no facility for creating imagery without technical aids. This is in significant contrast to sound, where humans have relatively equal facility for producing and hearing sounds as a means of communication among themselves.

Some of you probably grew up in the 50s and the 60s as did I. In the U.S., at least, Life Magazine played an interesting role in education. Whatever you thought of it, Life was a ubiquitous source of imagery. The pictures that Life provided were cut-and-pasted onto the classroom bulletin board of every classroom I was in until high school. The pictures arrived in the mail every week. Multiple copies, as many as you wanted. If you needed more, you could just ask a neighbor or a friend; lots of people had subscriptions and garages and attics provided a treasure trove of back issues. The pictures were almost literally free, so there was never any problem cutting them up and using them whenever imagery was needed. And there was almost always a large collection of imagery available, so long as you had access to the "archive."

The promise of multimedia is a return to that Golden Age. We have had electronic cut-and-paste of still images for quite some time now, but not a lot of access to the images. Networking promises to solve this (for a price – but the price will be small once it works) and to provide access to moving imagery. This will largely eliminate the distinction between real-time imagery and pre-computer imagery; all images will be pre-computed even if only just a little before they are viewed. The distinction between local and remote imagery will be lost; all images will go through the same multimedia pipeline just as if they came across a network. And compression techniques will wipe out the distinction between real and synthetic imagery; the two will be indistinguishable and many images will in fact be a blend of the two.

So where will this leave computer graphics?

It will still be there, but mostly as a producer of imagery on the network and as a tool for augmenting imagery obtained from other sources, both to customize it (to enhance its relevance) and to re-style it (to improve its fit with other images). Indeed, future multimedia systems, like existing text processing systems, will concentrate as much on style as on content (Beach & Stone, 1985).

The irony may well be that the very phenomenon that knocks computer graphics from its throne as the hottest thing in computing, that relegates computer graphics to a relatively isolated area of computer science research, will also be the force that finally brings the long-awaited promise of interactive graphics to education.

The power of multimedia lies in its ability to provide a powerful authoring tool that lets us appropriate imagery for the purpose of communication. Unlike computer graphics, which lets us create imagery, multimedia allows us to manipulate imagery. There is a world of difference. While creating imagery sounds like a good idea at first, anyone who actually does it realizes that it is very time consuming even with the best tools. If our goal is communication, we are better off using existing imagery to communicate our ideas or, if that fails, modifying existing imagery to our needs.

At roughly the same time there were two important advances made in the field of computing: Ivan Sutherland's PhD thesis *Sketchpad*, which set much of the the research agenda for computer graphics for a number of decades (Sutherland, 1965), and Doug Englebart's *NLS* (oN-Line System), which set much of the research agenda for multimedia and hypermedia for a number of decades (Englebart & English, 1968). Of the two, I think that Englebart, and the earlier vision of Vannevar Bush (1945), will be the more important for education and humanity in general simply because computer graphics is in the end just a set of technical tools for producing imagery whereas multimedia is a set of tools for using imagery.

References

Akeley, K., & Jermoluk, T. (1988). High performance polygon rendering. *Computer Graphics*, 22(4), 239-246.

Baecker, R.M. (1979). Digital video display systems and dynamic graphics. *Computer Graphics*, 13(2), 48-56.

Beach, R.J., & Stone, M. (1984). Graphical Style – Towards high quality illustrations. *Computer Graphics*, 17(3), 127-135.

Bush, V. (1945). As we may think. *Atlantic Monthly.*

Deering, M. (1992). High resolution virtual reality. *Computer Graphics*, 26(2),.195-202.

Ellis, S.R. (1991). Nature and origins of virtual environments – A bibliographical essay. *Computing Systems in Engineering*, 2(4), 321-347.

Englebart, D.C., & English, W.K. (1968). A research center for augmenting human intellect. *Proceedings FJCC*, Washington, D.C.: Thompson Books.

Foley, J.D., van Dam, A., Feiner, S.K., & Hughes, J.F. (1990). *Computer Graphics: Principles and Practice (2nd edition).* Reading, MA: Addison-Wesley.

Haeberli, P., & Akeley, K. (1990). The accumulation buffer: Hardware support for high-quality rendering. *Computer Graphics*, 24(4), 309-318.

IEEE (1985). *Proceedings of the 1st IEEE International Conference on Computer Workstations.* San Jose, California. Washington, D.C.: IEEE Computer Society Press.

IEEE (1988). *Proceedings of the 2nd IEEE Conference on Computer Workstations.* Santa Clara, California. Washington, D.C.: IEEE Computer Society Press.

Levinthal, A., & Porter, P. (1985). Chap – A SIMD graphics processor. *Computer Graphics*, 18(3), 77-82.

Maurer, H. (1992). Seminar in the Department of Computer Science at The University of British Columbia.

Myer, T.H., & Sutherland, I.E. (1968). On the design of display processors. *CACM* 11(6), 410-414.

Sutherland, I.E. (1963). Sketchpad: A man-machine graphical communication system. *Proceedings SJCC*, Baltimore, MD: Spartan Books.

van Dam, A., Stabler, G.M., & Harrington, R.J. (1974). *Proceedings of the IEEE*, 62(4), 483-492.

Watson, R. (1970) *Timesharing System Design Concepts.* New York, NY: McGraw-Hill

Acknowledgements

The preparation of this paper was supported by the Natural Sciences and Engineering Research Council of Canada and by the Media and Graphics Interdisciplinary Centre at the University of British Columbia.

SAFARI: an Environment for Creating Tutoring Systems in Industrial Training

J. GECSEI and C. FRASSON
Département IRO
Université de Montréal, CP 6128, Montréal, Québec, H3C 3J7, Canada
email: gecsei@iro.umontreal.ca

Abstract: Safari is a cooperative project involving universities, industrial partners and government. It aims at developing various architectures for industrial training, with different levels of complexity. The distinguishing features of the Safari environment are that (a) an attempt is made to represent knowledge at two levels: at the physical level corresponding to a simulated device, and the plan level; (b) tutoring is based on four instructional modes: demonstration, exploration, coaching and critiquing; (c) within every mode the development of progessively more complex prototypes is forseen, and tutoring in every mode involves the two levels of knowledge representation.

Creating tutoring systems remains difficult for various reasons: the different components of their architecture are not stabilized, the structure and type of knowledge to be used are of different granularity and complex to handle, and they require multiple expertise which is difficult to integrate [Fath *et al.* 1990, Lesgold *et al.* 1992].

The utility of tutoring systems having more or less intelligent capabilities is obvious. Most intelligent tutoring systems have been designed for academic domains such as mathematics, physics and computer programming. Although we do not lack teachers in such domains, there is still a well-known need for improving the quality of teaching in universities and colleges. However, there is an even more urgent need for computerizing industrial training. In todays global industrial competition, the skills required from workers are rapidly changing with the shifting technological environment, and employees must acquire new know-hows in limited time. The problem is compounded by a permanent lack of qualified instructors. For this reason, the research undertaken in the Safari project is focused on developing a tool set for the rapid construction of various tutors for industrial training.

Safari is a project under the auspices of Synergie, a programme sponsored by the Ministry of Technology and Science of the Governement of Québec. The main objectives of Synergie are (1) to enhance cooperative research and development between universities and industry, (2) to accelerate the product development cycle, (3) to facilitate the transfer of knowledge between research establishments and industry, and (4) to educate highly qualified professionals in the domain. Safari involves four Québec universities, two industrial partners and a government agency. The industrial partners are Virtual Prototypes Inc., providing a software package VAPS and Novasys Inc., a consulting firm specialized in the AI and training domains. VAPS (Virtual Applications Prototyping System) is a high-quality commercial interface-building and simulation system, used in many areas (such as industrial design and evaluation of airline cockpits).

The Safari team includes seven professors conducting research on a part-time basis : C. Frasson (project leader), G. Gauthier, J. Gecsei, G. Imbeau, M. Kaltenbach, S. Lajoie and B. Lefebvre; about 20 M.Sc and Ph.D students, two full-time programmers and several engineers provided by the industrial partners. The project time frame spans 4 years (1993-96).

Building reasonable tutoring components

The main objective of Safari is to develop a *methodology* and an *environment* for the *creation of tutoring systems to be used in professional training.* The focus is on teaching mostly procedural knowledge concerning the operation of devices such as medical instruments, manufacturing robots, consumer appliances, control instruments, aeronautical instruments, etc.

The basic idea is *to add a tutoring component on the top of device models (microworlds) built in VAPS.* This permits the use of models written in VAPS ("virtual instruments"), instead of the real devices, for training and practice. Thus the expensive machinery can be kept for its actual purpose and damages from incorrect manipulations during learning can be prevented. Safari can be seen as a value-adding component attached to the VAPS package, making it into an integrated toolkit for the creation of device models together with a corresponding computerized training course on how to operate the device. Since a large base of VAPS device models are already available, these can be conveniently used to validate the Safari approach.

In order to remain within the bounds of practicality, we limited a priori the target applications to devices which have the following characteristics:

- They permit the definition of a set of clearly distinguishable interaction procedures (e.g. through control panels, switches).
- The device functions are decomposable into a number of well-defined *tasks*, each task corresponding to a meaningful operation in terms of the device's main purpose.

These characteristics exclude very complex equipment such as nuclear reactors (although parts of it may be covered) and devices whose operation requires hard to enumerate, "look-and-feel" actions such as handling certain manufacturing machines, or performing surgery.

Tutoring systems created in the Safari environment will be capable of :

- tutoring workers in an industry (factory, hospital) through *real tasks* recreated on device models using the VAPS software,
- constructing *models of individual users*, reflecting the user's reasoning style and level of knowledge,
- dispensing *individualized and adaptive tutoring* based on the user model,
- diagnosing the actions of a trainee when asked to perform a task on the virtual instrument, and
- composing an entire curriculum for teaching the operation of the device, taking into account the learner's expertise and style of learning.

Some distinguishing features of Safari are that

- it involves contractual cooperation between the industrial and university partners who are committed to a binding schedule including the development of commercially viable products,
- the proposed methodology is applicable to a large class of devices and domains, not only to a single device type,
- it is based on an existing software package VAPS, and
- it enhances learning by using multiple representations of knowledge, and by encouraging the learner to pass between them.

We do not aim exclusively at deeply intelligent tutoring; this would be somewhat unrealistic, given the limited success and the problems such systems have even within narrow application domains. Instead, Safari provides a combination of some existing tutoring approaches to build easily and rapidly a tutoring system for a given device.

A realistic training sequence

Throughout the project time frame we use the following functional paradigm. In Safari, tutoring functions (modes) are based on the observation that the natural cycle in which most people acquire a given skill is by first *observing* someone's demonstration of the skill, then freely *experimenting* with the device in question (given the availability of the device, and that such experimenting is not hazardous), then *executing* precise tasks (assignments) in terms of the device functionalities under the guidance of an expert, and finally by *describing* the learned skill in an abstract form (which can be used as a basis for a critical examination by an expert, or for communicating to other people).

Somewhat simplifying, in our approach this translates into four distinct *tutoring modes*: demonstration, free exploration, coaching and critiquing. These modes are presented to the learner through an interface which is similar in all modes, and which displays two major windows: the Device Window showing the simulated device, and the Plan Window showing an abstract representation of a task called *task graph*. In this arrangement the learning process involves two related views of the learning space, encouraging the learner to develop his knowledge simultaneously on the concrete and abstract level. This method has its origin in the "PIF" approach to tutoring, described in [Frasson *et al.* 1992]. PIF supports three interrelated "worlds" (i.e. levels of representation): the Physical, Intentional and Functional. In Safari we retained the P and I worlds, corresponding to the Device and Plan windows.

In *demonstration mode*, the student can observe the execution of various tasks applicable to the device, somewhat like a videotaped demonstration. However, in Safari the demonstration is based on the VAPS model, executing automatically a task graph (TG) which had been edited by a human expert. A *demonstration scenario* is a TG augmented by hypermedia comments and explanations attached to certain elements of the TG. The student can navigate in the scenario by stopping, repeating, jumping to an arbitrary point (i.e. by clicking on an action in the TG), and by asking for hypermedia explanations. However, he *cannot* interact directly with the simulated device, only with the scenario. As the demonstration progresses, actions in the TG are highlighted along with the corresponding simulated manipulations of the device (such as operating the controls).

The purpose of *exploration mode* is to permit direct interaction with the microworld (virtual device). Here the learner is allowed to freely experiment with the device, to observe the consequences of his actions in a realistic environment, but without the inconveniences of experimenting with the real device [Fath *et al.* 1990]. The system may attempt to determine which task the learner is trying to execute; in this case the TG of the conjectured task is gradually displayed in the task window according to the learner's progress in manipulating the device. Exploration can also be done within the context of a given task; then the student is allowed to interact with either window, with automatic highlighting of the corresponding actions in the other window.

In *coaching mode* the student is asked by the system to perform a given task or to solve a given problem scenario (=sequence of tasks) by interacting only with the Device Window. Initially the correct TG is hidden; it is gradually revealed as the learner's solution is being built. The process is supervised by a coaching component, essentially analyzing the difference between the correct solution(s) from the knowledge base and the learner's construction, and generating appropriate coaching interventions.

The fourth tutoring mode is *critiquing*, where the learner has to prove his knowledge by constructing a solution in the form of a TG for a given problem. The learner can "experiment" in the device window as in exploration mode, and build the solution in the Task Window. In this way he is encouraged to pass frequently between alternative representations in the two windows; in other words, the student has to prove not only that he can correctly execute a task, but also that he *understands* (to a certain degree, of course) what he is doing. By requesting the student to construct a TG, he is effectively asked to express his knowledge in an abstract form. When the proposed solution is complete, a critiquing component (similar to the coach) will comment on the solution.
At any time during work in the above tutoring modes, the student can ask for help (non-contextual) or contextual advice.

Evolution

As shown in Figure 1, the four tutoring modes are considered as evolving in time from minimal versions to more sophisticated. In this way we can keep the project consistent over a period of time, and by producing a series of prototypes of increasing complexity we can exploit the experience learned on the way. We firmly believe that a well functioning complex system can be created only by evolving through simpler versions. The evolution of prototypes can be roughly divided in three phases according to the complexity of the knowledge structures involved. In the following, we briefly characterize these phases.

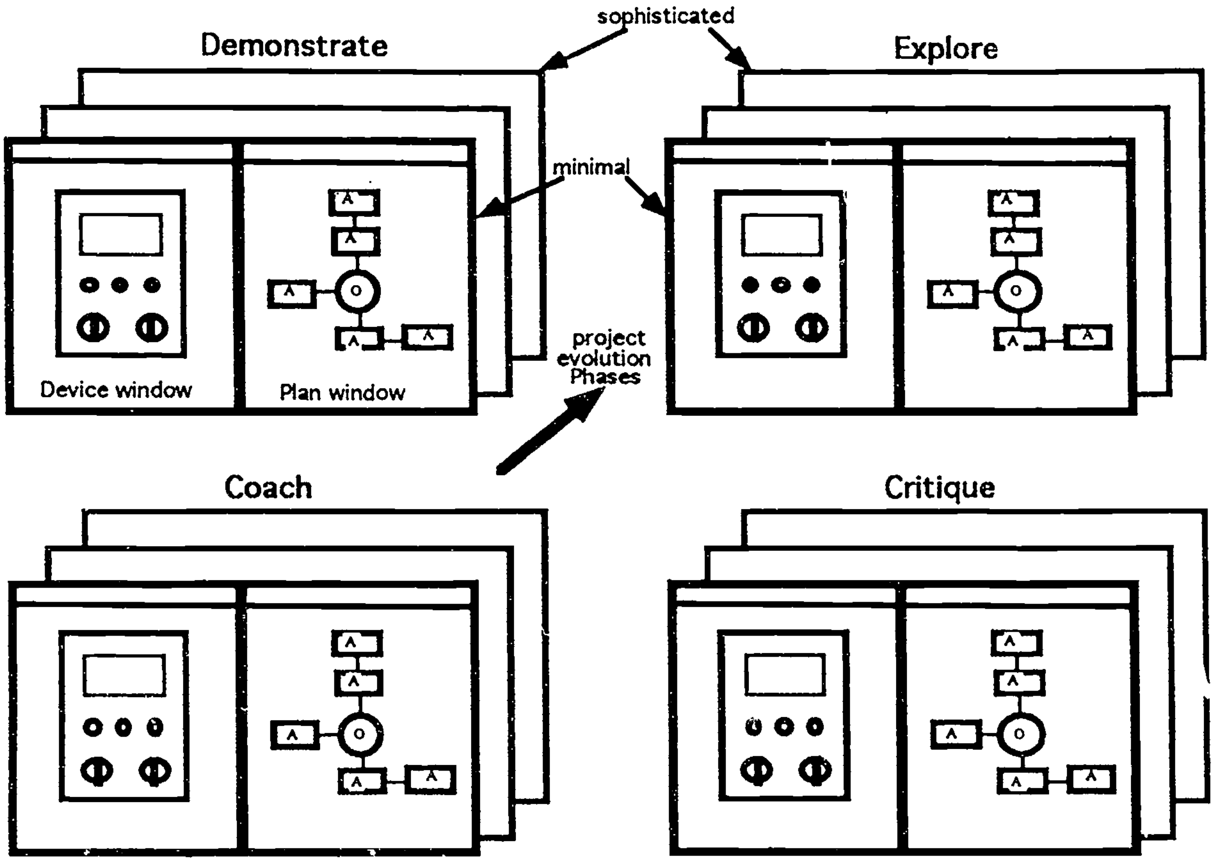


Figure 1. Safari tutoring modes and evolution.

Phase 1: In Phase 1, task graphs are equivalent to flowcharts containing two kinds of nodes: Action (A) and Observation (O) nodes [Mittal *et al.* 1988]. Actions are operations to be executed by the user, Observations serve as decision points where the following action depends on some observable behavior of the device. There is one TG per task, and all tasks can be clearly distinguished. The knowledge base consists of the set of corresponding TGs. A demonstration is a pre-recorded manipulation of the VAPS virtual device; in exploration mode the student can interact freely with the model. Coaching and critiquing are based on overlay-like comparisons of the student's input and the correct TG from the knowledge base. Even so, some nontrivial comments can be generated such as "Your solution is almost correct, except that you inversed actions A and B."

Phase 2: Here, we will take into account some factors that make the model of Phase 1 overly simplistic. First, in many cases there is more than one way to accomplish a task. These ways may be equivalent, or can have different attributes in terms of efficiency, cost, etc. We have to deal with this multiplicity in terms of efficient knowledge base representation and retrieval mechanisms. Second, the VAPS-based virtual device may be insufficient for efficient coaching and critiquing interventions which may need, for example, causal explanations. To achieve this we will introduce a *cognitive model* of the device, complementary to the lower level VAPS model. In Phase 2, the internal representation of knowledge (tasks, cognitive device model) will be based on concepts and relations integrated in an object structure. A simple student model (overlay) will permit to individualize tutoring interactions.

Phase 3: Here, tutoring will involve other types of device knowledge, such as structural, functional and maintenance. In further experimenting with the coaching and critiquing modes, we will emphasize the idea of having the learner gradually *create* and *refine* his knowledge simultaneously on two levels (instead of just showing a task and asking him to prove his knowledge). The student model will permit to infer the student's reasoning from his actions and adapt the coaching interventions accordingly. We will attempt to make certain types of knowledge transferable to new situations [Wielinga *et al.* 1992]. Figure 2 shows the main functional blocks of the Safari system.

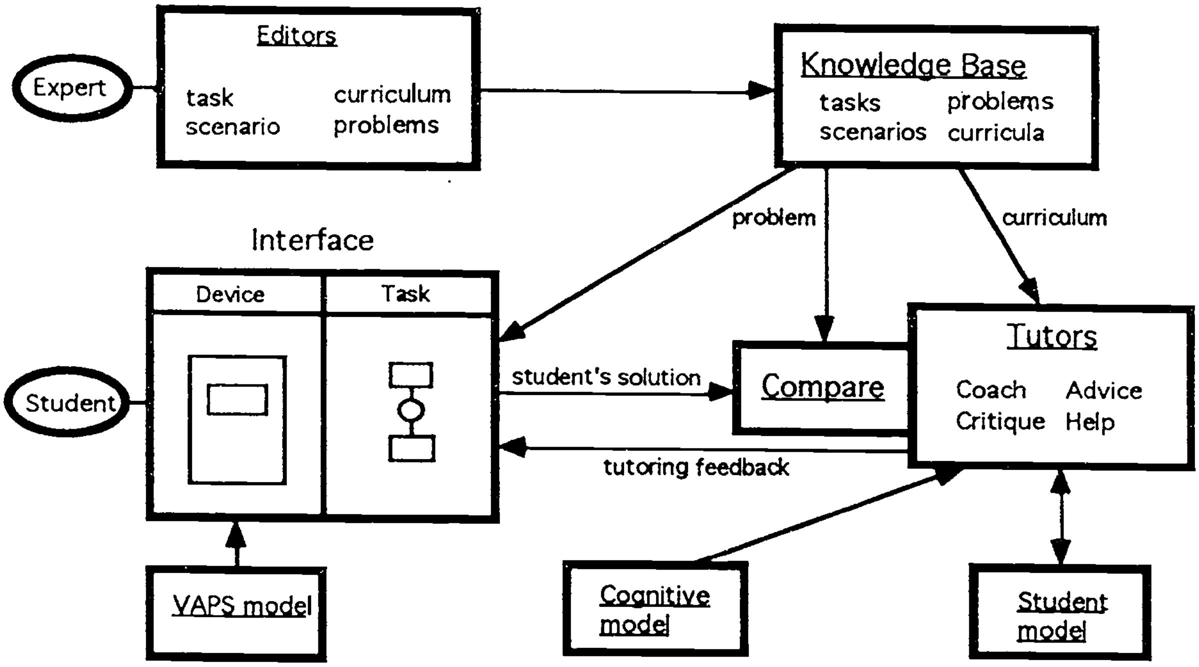


Figure. 2. Safari functional architecture.

Each component will gain in complexity in successive phases. The architecture is organized in a way that makes it possible to implement a layered development of the components. Each component contains a base to which a more complex module can be attached. Prototype implementation is under way on Sun and Silicon Graphics workstations. The two main programming languages used are VAPS (used for the virtual device and the demonstration editor) and Smalltalk (used for the other components).

Implementation: example

One of the first devices used for Safari prototyping is from the medical domain: the Flo-Gard® 6201Volumetric infusion pump fabricated by Baxter Healthcare Corporation. The device, used mostly in emergency rooms, can deliver a variety of fluids over a broad range of infusion rates. The rates can be programmed in different ways, such as over a fixed period of time, or with constant rate until a given volume has been reached, or as a sequence of infusion programs. The machine can control two independent simultaneous infusion processes. During an infusion, a number of special events can occur, such as lack of fluid in the reservoir, or an air bubble in the infusion tube. The trainees (health technicians and nurses) have to learn the basic operating procedures, as well as to handle special events. The Device Window on the left of Fig. 3 shows the front panel of the pump, as simulated in VAPS.

A cognitive analysis of the learning situation resulted in some 20 tasks which are expressed as task graphs. Related tasks are regrouped in curriculum issues; these were in turn classified according to their degree of difficulty (novice, intermediate, advanced). The righthand side of Fig. 3 contains a scenario editor used for creating a demonstration of a given task. This type of demonstration consists of actions (pre-recorded manipulations of the virtual device, represented by trace-footstep icons), and written and spoken comments (pre-recorded files represented by appropriate icons). The editor permits to arrange the icons in a desired presentation sequence. When an action is being replayed, the corresponding manipulations of the device controls (switches, keys) are visualized by the hand icon moving to the appropriate places, and by highlighting the activated controls.

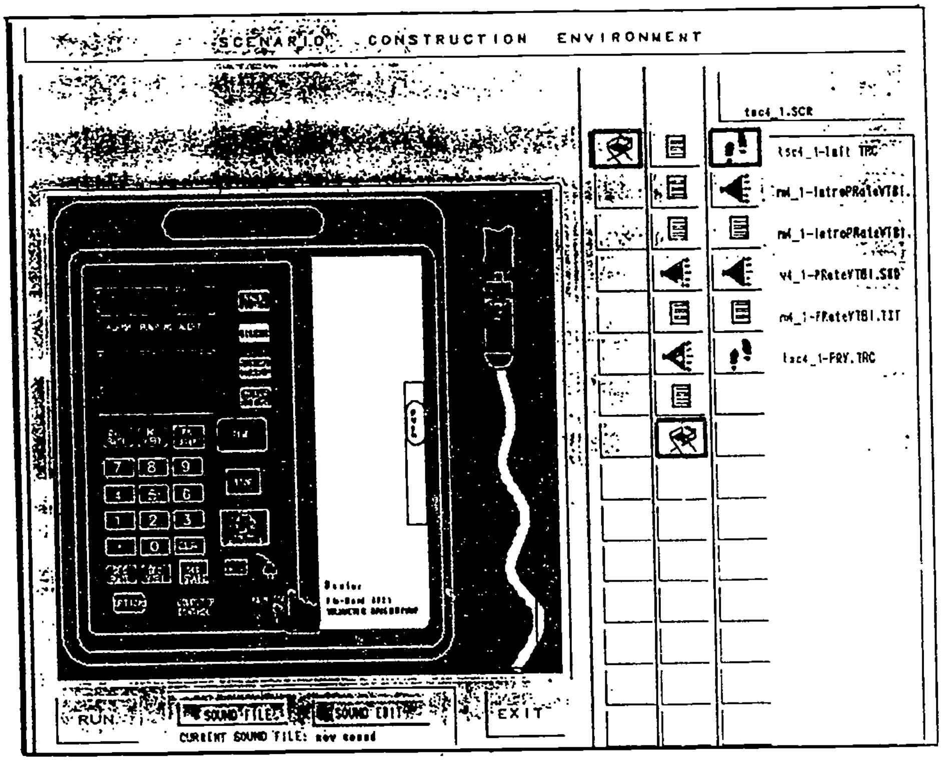


Figure 3. Infusion pump demonstration scenario editor.

Acknowledgements

A number of students contributed to the work reported here. We are particularly indebted to J.-Y. Djamen, A. Kabbaj, A. Mayers, R. Bouchard, A. Kengne, L. Quirion, Ch. Abadie, L. Duperval, R. Nkambou, Tang Ho Le, D. Phalp and N. Sandrasegaran.

References

Fath, J.L., Mitchell, C.M., Govindraj, T. (1990) *An ICAI Architecture for Troubleshooting in Complex, Dynamic Systems,* IEEE T-SMC, 20,3,pp. 537-558.

Frasson, C., Kaltenbach, M., Gecsei, J., Djamen, J-Y. (1992) "*PIF: an iconic intention-driven ITS environment*", in Frasson, C., Gauthier, G., McCalla, G. (eds), "Intelligent Tutoring Systems", Lecture Notes in Computer Science, vol 608, Springer-Verlag.

Lesgold, A., Lajoie,S., Bunzo, M. and Eggan, G. (1992)*SHERLOCK: A Coached Practice Environment for an Electronics Troubleshooting Job,* pp. 201-238 in "Computer-Assisted Instruction and Intelligent Tutoring Systems: Shared Goals and Complementary Approaches", J.H. Larkin and R.W. Chabay (Eds.), L. Erlbaum Associates.

Mittal, S., Bobrow, D.G., de Kleer, J. (1988) *DARN: Toward a Community Memory for Diagnosis and Repair Tasks,* Ch. 4 in: J. A. Hendler, Ed., Expert Systems: The User Interface, Ablex.

Wielinga, B. J. et al. (1992) *KADS: A Modelling Approach to Knowlegde Engineering,* Knowledge Acquisition, 4(1).

Integrated Multimedia in Distance Education

J.M. Greenberg
Academic Computing Service
The Open University
Walton Hall
MK7 6AA
United Kingdom
j.greenberg@open.ac.uk

Abstract

The British Open University is a world leader in distance education and a range of media has been used successfully in its teaching programme. The University has been looking at ways of creating integrated multimedia learning environments for students for a number of years. The growth of the University's Home Computing Programme and advances in multimedia technology will make these types of environments a reality for Open University students before the end of the decade. A number of projects have already addressed the technical issues and several new projects have been undertaken to look at the potential of multimedia in distance education. One particular development planned for completion in 1994 will result in a multimedia learning environment built around Homer's epic poems and their archaeological contexts.

Introduction

The Open University delivers distance education to over 200,000 students each year. At the heart of most courses is a set of specially-written textbooks or workbooks. Additional material may also include special equipment lent to the student for practical work, audio and videocassettes and computer software. Many courses include radio and television programmes on the national BBC networks. The University offers students a regionally based teaching and counselling support system through its thirteen regional centres. Some courses include one-week residential schools which are usually held in the summer months at other universities or conference centres in the United Kingdom. There are also shorter residential schools, part-week and weekend for other courses.

The University has been a multimedia institution since its inception and in addition to vast amounts of printed materials, currently ships to students in a single year: over 34,000 special equipment kits, around one million audiocassettes, over 100,000 videocassettes and over 350,000 computer disks. As in most other higher education institutions, there has been a rapid growth in the use of the personal computer as a teaching media in Open University courses. A Home Computing Programme was launched in 1988 which required students taking certain courses, to provide their own access to PC compatible machines meeting a minimum specification. This programme now includes 18 courses and supports over 20,000 students. While the current minimum specification does not fully support new multimedia technologies, it is being upgraded for all new courses from 1995. This new minimum specification will support these technologies and will also include a communications facility. It is important to note that the issue of student access to computing equipment is a fundamental problem associated with the introduction of computer-based teaching to distance education courses.

The University's Residential School Programme currently offers computing activities to over 40,000 students and in 1994, over 900 computers will be installed for student use. These schools allow the

University to set up computer laboratories more common to traditional universities. They are often used as a test-bed for new technologies and activities based on interactive videodisc systems and CD-ROM have been available to students for a number of years.

Non-Integrated Multimedia in Distance Education

Every Open University course is produced by a multi-disciplinary team and depending on the choice of teaching media, normally includes Academics, Software Designers, Graphics Designers, Editors, BBC Producers and Educational Technologists. All course material is considered by the team and a number of drafts of printed material, versions of software and scripts for audio and video material are produced.

While it is true to say that the Open University is a multimedia institution, the integration of different media types on most courses takes place through print-based material. Students are provided with printed course guides and a study calender which recommends a weekly schedule. In a course which provides a wide range of media, the logistics of home-based study have to be taken into account in developing these course guides. Many students have spouses and children and are unlikely to have a study area equipped with a computer, video and audio equipment. So their use of the different media has to be planned around their family's activities and in many cases, the real integration of the media takes place in the student's head.

Early attempts to overcome these difficulties involved the use of audiocassettes to assist the student's study of particular printed material. This idea was extended to computer-based activities in 1988 when a course in computational mathematics developed a number of teaching packages. Each package was intended to reinforce specific teaching points and was accompanied by an audiocassette, whose commentary was scripted to match the software. The packages were produced jointly by Academics in the Faculty of Mathematics and Computing, BBC Producers and Software Designers from the University's Educational Software Group.

The Integration Process

While the University continues to develop computer-based activities for use at residential schools which exploit multimedia technologies, new projects are underway to develop integrated learning environments and establish their effectiveness, particularly in home-based study.

Project Choice

As in most higher education institutions, the Open University is committed to a specific programme of course development in a climate of fierce competition. A proposal to produce a course based on new multimedia teaching material would not be favourably received at this time, given the student access issue and the uncertainty over the quality of a course of this kind. It is more appropriate to find an existing course which lends itself to an integrated multimedia approach and establish that a multimedia presentation enhances the student's learning experience.

The course A295 *Homer: Poetry and Society* which was first presented in 1993 was chosen because from the outset, it was conceived as a highly integrated course in which video, audio and print would work together in a complimentary way. This format lends itself to a technology that permits immediate access to any of these media at a single source. The course contains a number of areas that are difficult to teach using non-integrated media, for example; showing students how to use and learn from archaeological site plans or searching the texts of the poems in a structured way.

The Project

The 'Homer Project' is based on some of the material from the course A295 *Homer: Poetry and Society*. The course as a whole consists of the following components:

Set Books:	Homer's *Iliad* and *Odyssey* and a commentary text for each.
Essays:	15 essays from *Homer: Readings and Images*
Video:	2 x 120 minute videocassettes
Audio:	4 x 90 minute and 1 x 60 minute audiocassettes
Learning Guides:	5 printed guides which contain all the pedagogical apparatus relating to and integrating the above items.

The project involves converting the material associated with Learning Guide Two into a multimedia presentation. This material consists of about 750,000 words of text, 96 minutes of audio and 100 minutes of video.

The aim of the project is to test and report on the success of multimedia as a humanities distance-teaching medium; to identify practically its limitations both in terms of its production and presentation; and to assess the ways the medium works with traditional media. The project will provide basic information for the University as it plans for the introduction of multimedia technologies into its next generation of courses.

The Production Team

The 'Homer Project' has brought together a multi-disciplinary development team which is based around members of the original course team. The two BBC Producers and BBC Graphics Designer produced the audio visual material in the course and the Academic Consultant was a key academic member of the course team. The project is being managed by the University's Educational Software Manager and a Senior Software Designer and Software Designer are responsible for software design and implementation. The original course Editor has been seconded to the project for a year and is contributing both academically and editorially. An Educational Technologist from the University's Institute of Educational Technology is making a major contribution to the pedagogical design of the material and is responsible for developmental testing and evaluation.

The team operates in a similar fashion to a traditional Open University course team, with all members contributing to and commenting on all aspects of the project. A major difference however is that the Producers and Graphics Designer are playing a much more central role in the design process than they normally would in a non-integrated multimedia course.

The Teaching Strategy

The basic principle argued for in the project and taught in the existing course, is that the *Iliad* cannot be fully understood and appreciated without considering the relationship between the poem and its possible material contexts provided by examination of archaeological sites, by looking at landscape, topographical details and so on.

The project team decided to develop two strands, one based on the archaeology and the other based on the literature. These strands are interwoven throughout a central spine or study route through the course material. The BBC Producers, in consultation with the Academic Consultant were responsible for developing all of the archaeological based teaching material, while the Editor and academic consultant concentrated on the literary based material.

The study route consists of six weeks of study and each week has a number of investigations to be undertaken by the students. Each weeks' study concludes with a recapitulation of the main teaching points and a target milestone in the student's understanding of the material. Each investigation draws on a range of multimedia materials which are described below. The study route is intended to encourage a guided discovery form of study rather than one based on dialogue, which is the more traditional computer assisted learning approach. It is hoped that this induces in students a sense of being in control of the learning process, rather than waiting to be told what to do next.

Producing the Integrated Learning Environment

A learning environment has been designed which allows the student to progress along a prescribed study route on a weekly basis. While all of the multimedia teaching materials are integrated into the study route, they can also be accessed by the student independently, with their own user interface. These materials are categorized as follows:

Poems:	The complete text of the *Iliad* and the *Odyssey*
Library:	A number of essays and companion texts to the *Iliad* and the *Odyssey*
Sites:	A number of site plans and audio visual material associated with Troy and Mycenae
Skills:	Activities to improve archaeological and literary skills
Museum:	A number of photographs of relevant artifacts held in museums and audio visual material linking them to Troy and Mycenae
A-Z:	A chronological table relevant to the period covered in the poems

All text material can be accessed with full text retrieval tools and students have access to a notebook facility. This allows them to make notes and link these to specific quotes from the text material. Within the investigations there are textual discussions with hyperlinks to the poems.

Extensive use has been made of stills with audio commentaries and this has significantly reduced the amount of video necessary to make specific teaching points. The concept of a "vision bite", where use is made of very short video sequences is not compatible with a non-integrated multimedia presentation, but quite possible when the media becomes fully integrated.

Following the University's student computing model, the learning environment runs under Microsoft Windows. While most educational software developed for Open University students uses software tools developed in-house and written in C++, these tools do not yet offer full multimedia support. Given the time scales imposed on the project, it was decided to use commercial tools and development work was carried out using Asymetrix TOOLBOOK and Microsoft Multimedia Viewer 2.0.

The delivery platform is a high specification PC compatible with 16-bit multimedia audio card, Intel ActionMedia II delivery board and dual speed CD-ROM drives. Video files are stored in Intel's Indeo Video format and although the software will run without the ActionMedia delivery boards installed, they are desirable to achieve the highest possible video quality.

Copyright

All of the material currently used by students in the course has been cleared for copyright in its present form. Putting the same material into a different format has meant that all of the copyright clearances have had to be renegotiated. While in many cases this has just had a cash cost associated with it, some holders of copyright have refused to grant clearance. The University is fortunate to have a Rights Department to deal with this problem, but it is an issue which will continue to be fundamental to the development of multimedia educational materials.

Developmental Testing

As each component of the learning environment is completed, individuals interested in the material have been used to test it. This testing takes place with the Educational Technologist in attendance and feedback is provided to the software developers or to the project team, if fundamental design questions are raised.

The completed multimedia material is being distributed to about 20 students who have already taken the course. They are being selected to cover a range of study success and being asked to restudy the material in Learning Guide Two, using the multimedia materials. The course team do not wish to see the multimedia parts of the course developmentally tested for credit at the outset. If the initial developmental testing indicates that students will not be disadvantaged, then a developmental testing programme for credit may be built around the multimedia elements of the entire course.

The University's Institute of Educational Technology will be carrying out a project evaluation after students complete the study period. The Educational Technologists acknowledge that it is not clear how one assesses the relative educational effectiveness of a non-integrated and integrated multimedia presentation of the same material. They hope that through this project, guidelines may emerge for future courses.

Conclusions

The Open University has recognised the difficulties that a number of its course teams face in presenting their courses through non-integrated multimedia learning environments. Projects such as the one described in this paper are intended to establish that integrated multimedia learning environments will considerably enhance the learning experience of students in distance education. Some 20,000 students are currently studying courses which require them to have access to a personal computer. From 1996, a considerable number of students will have computers which support multimedia applications and they will quite reasonably expect the University to provide them with integrated multimedia teaching materials.

The integration of multimedia learning environments into a specific course is labour intensive and while this paper is being written before the report on the 'Homer Project' is complete, it is already clear that such developments are going to be very expensive in the short term. We can only hope that once these developments become a normal part of the University's course production process, integrated multimedia will offer benefits to the institution as well as its students.

Acknowledgements

The author would like to thank the 'Homer Project' development team: Producers Tony Coe and Mags Noble, Graphics Designer Robin Arkle, Academic Consultant John Purkis, Course Editor and Academic Consultant Peter Wright, Educational Technologist Diana Laurillard, Senior Software Designer Ian Every and Software Designer Colin Rutledge.

VIRTUAL CLASSROOM PLUS VIDEO: TECHNOLOGY FOR EDUCATIONAL EXCELLENCE:

Starr Roxanne Hiltz and Murray Turoff
New Jersey Institute of Technology, Newark NJ 07102 US

ABSTRACT: The Virtual Classroom™ is a teaching and learning environment constructed in software, which supports collaborative learning among students who participate at times and places of their choosing, through computer networks. This paper describes some of the authors' experiences utilizing Computer Mediated Communications structured to create a Virtual Classroom (VC). A project which is creating an entire degree program, the B.A. in Information Systems, via mixed mode delivery, (VC combined with video), is then described.

Computer-mediated communication systems (CMCS), especially when enhanced to create what we refer to as a Virtual Classroom,[TM] can make significant improvements in both access to and the quality of education. Currently over 80 programs worldwide are known to be offering courses partially or completely via CMC (See, for example, Harasim, 1990; Harasim, Hiltz, Teles & Turoff, 1994; Hiltz, 1986, 1994; Mason & Kaye, 1989; McCreary & VanDoren, 1987; Paulsen & Rekkedal, 1990; Rice & Case, 1983; Weedman, 1991; Welsch, 1982). The sophistication and flexibility of software structures for supporting distance education vary widely, from simple electronic mail systems to conferencing systems that have been specially enhanced to support classroom-like experiences, particularly group discussions and joint projects.

Generically, the Virtual Classroom is a teaching and learning environment located within a computer-mediated communication system. Rather than being built of steel and concrete, it consists of a set of group communication and work "spaces" and facilities that are constructed in software. Thus it is a "virtual" facility for interaction among the members of a class, rather than a physical space. Specifically, the Virtual Classroom™ is NJIT's trademarked name for a version of its Electronic Information Exchange System (EIES2) with special software structures designed to support collaborative learning. Participation is generally asynchronous; that is, the Virtual Classroom participants may dial in at any time around the clock, and from any location in the world accessible by a reliable telephone system. The fact that the educational process is asynchronous means each student may engage in more reflective thinking before having to answer or discuss issues. Some of the Virtual Classroom facilities and their traditional classroom analogies are summarized in Figure 1. For example, interaction in the form of "electures" (electronic lectures) and plenary discussions takes place in a main "class conference," which is like the classroom or lecture hall. For special activities such as a debate or small group work, other conferences can be established (like moving from a lecture hall to breakout rooms). Private conversations, the equivalent of hallway conversations or office hours, take place via "messages."

Note that messaging and conferencing serve very different objectives. Trying to hold group discussions and develop a shared database of the discussion is impossible in a message system, and cluttering up conferences with a lot of transitory material better sent in messages interferes with the flow of the discussion. Being able to utilize a key word index for finding material such as assignments in the conference and having a way of organizing the discussion threads are extremely critical to the use of the conference transcript as a database.

The special features added to a basic CMC to support collaborative assignments have been integrated into EIES 2, as "Activities," which are executable programs attached to an ordinary conference comment. Rather than automatically receiving everything that has been entered by any participant, as with comments, participants choose to do the activities only when they are ready, and explicitly give a command. A record is kept of ***done*** and ***undone*** activities for each conference member, and a review choice helps users to keep track of what they have accomplished.

Figure 1
Communication Structures in the Virtual and Traditional Classrooms

Computer Facility	Utilization	Physical Analogy
Private Conferences	Class Discussions & Lectures Student working groups Student Lounges	Classroom Study Groups
Public Conferences	Teacher Lounges Shared material (document data bases)	Coffee Houses
Messages	Student to student Teacher to student Transitory material	Office hours "Hallway" conversations
Notebooks and personal files	Composition facilities	Work book
Membership Status	Who has read and done what assignments (tracking)	Visual Presence
Binary File Attachments to Comments	Diagrams, Spread sheets, etc.	Sharing of PC software results
Anonymous Signatures	Encouraging Self-disclosure and experimentation Presenting mistakes Game and role playing	Impossible in face-to-face classroom
Membership Directory	Finding members by common interests	Clubs, interest group formulation
ACTIVITIES:		
Question/Response	Forces independent thinking and active participation	Face-to-face discussion questions
Selection	Manage distribution of unique assignments	Circulate sign up sheets
Document	Self selection of pieces and parts of long document	Printing press and copy machines
Exam	Time controlled question set	Written exam
Gradebook	Access to student grade record	Asking instructor
Voting	Voting for lists of alternatives	Voice votes or show of hands, ballots.

While students may access only their own records of done and undone activities, the instructor can review the Activities status of any of the students. The instructor can require activities to be done in certain sequences and declare whether they are required or optional. Activity types include:

* **QUESTION/RESPONSE**, which is the most frequently used. One or more questions for response by other conference members are contained in the main conference comment. Each person MUST ANSWER BEFORE SEEING THE RESPONSES OF OTHERS. This is very important for making sure that each person can independently think through and enter his or her own ideas, without being influenced by responses made by others. Alternatively the author may set it up so that participants cannot see other responses even after they answer, until the author "opens" the responses for viewing. This might be done for an essay-type quiz, for example.

* The electronic **GRADEBOOK** allows students to see not only their own grades and averages at any time, but also averages for the class as a whole. When grades are added, notifications are automatically sent to the members to alert them to the availability of new grades.

Collaborative Learning

Collaborative learning is defined as a learning process that emphasizes group or cooperative efforts among faculty and students. It stresses active participation and interaction on the part of both students and instructors. Knowledge is viewed as a social construct, and therefore the educational process is facilitated by social interaction in an environment that facilitates peer interaction, evaluation and cooperation (Bruffee, 1986; Johnson, 1981; Johnson & Johnson, 1975).

The Virtual Classroom is an environment that facilitates collaborative learning -- among students, between students and instructors, among teachers, and between a class and wider academic and nonacademic communities. It also supports independent learning and generative, active learning techniques that are self-paced by each participant. For distance education students, the increased ability to be in constant communication with other learners is obvious. But even for campus-based courses, the technology provides a means for a rich, collaborative learning environment that exceeds the traditional classroom in its ability to "connect" students and course materials on a round-the-clock basis.

An example of a collaborative learning strategy applied in the VC that is included in most courses is the "seminar" type of interchange in which the students become the teachers. Individuals or small groups of students are responsible for making a selection of a topic (usually from a list provided by the instructor as a Selection Activity); reading material not assigned to the rest of the class; preparing a written summary for the class of the most important ideas in the material; and leading a discussion on the topic or material for which they are responsible (usually via a Response Activity).

Seminar-style presentations and discussions are thus an example of a collaborative learning activity that is often difficult in the Traditional Classroom (TC), but which tends to work very well in the Virtual Classroom environment, even with fairly large classes of undergraduates. Other examples of collaborative learning strategy in the VC include debates, group projects, case study discussions, simulation and role-playing exercises, sharing of solutions to homework problems and/or answers to review questions for exams; and collaborative composition of essays, stories, or research plans.

Modes of Use of the Virtual Classroom

CMC technology can be utilized in many different modes to support education and training:

* As an adjunct to a regular face-to-face course in order to improve the communications between the students and the instructor, as well as to improve the communications among the students.
* As a mechanism for providing communications in a remote course where the students receive the lectures via video (by direct broadcast or by recorded tapes).
* As a total means of delivery, without any other communication mode.

In general, the more the course is oriented to teaching basic skills (such as deriving mathematical proofs), the more the lecture is needed in some form as an efficient means of delivering illustrations of skills. However, the more the course involves pragmatics, such as interpretations of case studies, the more valuable is the CMC mode of delivery. CMC is the ideal technology for extending the ability of students to discuss material and to work in collaborative groups as an integral part of the learning process. It is also the ideal technology for extending education or training to those segments of the population that have difficulty in taking the time to attend face to face sessions. Our initial studies (Hiltz, 1993) showed that mastery of course material in a variety of online courses was equal to or better than that in traditional courses, and subjective satisfaction was higher. However, for totally online courses, it is recommended that students be at the sophomore level or higher, or be screened to exclude freshmen with poor study skills.

Some of the courses incorporating video for "lectures" use standard public television courses, such as "Discovering Psychology," produced by PBS. Most video segments are filmed by NJIT in its "candid classroom" and then distributed to remote students on videotape, or via broadcast on a cable channel or satellite. In all video variations, the Virtual Classroom is used for all assignments and additional discussions.

Initial experiments with the mix of Video for "lecture" type material and VC for discussion and collaborative assignments worked so well that we proposed to deliver an entire degree program this way, and were fortunate enough to receive funding from the Alfred Sloan Foundation that has enabled us to do this.

Virtual Classroom + Video: The Degree Program

This project will develop, offer, and assess the effectiveness of an undergraduate major in Information Systems delivered via Virtual Classroom plus video, to attain five objectives:

* Faster progress towards the undergraduate degree, by facilitating self-paced learning and solving major educational logistics problems;
* Improved quality of learning through the increased collaborative learning and faculty-student interaction facilitated by computer conferencing;
* Increased access to educational opportunities for working adults or those trying to re-enter the work force, particularly women;
* Formative and summative evaluation of the effectiveness of this media mix used in different ways, for attaining the above objectives;
* Dissemination of the successful techniques and materials to other institutions.

Access: Overcoming Logistical Problems

The majority of NJIT's students are "first generation" college students, who must work while attending school and who commute rather than live on campus. They are faced with overcrowded classes with sections that fill up early and then become closed (due to budget cuts that laid off staff and raised class sizes). They are also faced with degree programs that because of their technological nature, have extensive sequences of courses that build on one another and must be taken in order. If a student is closed out of a course one semester or must withdraw, he or she might lose a whole semester waiting for another opportunity to take this course which is a pre-requisite for subsequent courses.

In addition, many of the courses are offered more slowly than would be possible for the better prepared or more motivated students to proceed. The "mass lectures" that have emerged as a result of budget cuts tend to slow down to the pace of the "lowest common denominator" and frustrate the better students. Both video and VC allow "fast forward" and "replay" self-pacing.

By using a combination of video (tape and/or broadcast) plus Virtual Classroom to deliver courses, students may more easily fit them into busy lives as employees and family members as well as students. Since the vast majority of NJIT students work 20 hours a week or more, and many work 40 hours a week or more (in order to pay their own educational expenses and/or to help support their families) it is very difficult for them to fit courses they need to graduate into their schedules. By offering distance education sections, scheduling difficulties can be alleviated. Students can graduate in fewer calendar years because they can more easily fit in all the courses they want.

In addition to the asynchronous nature of the VC helping to overcome course enrollment and completion difficulties, we are trying several innovations that may better facilitate self-pacing and improved rates of progress towards the degree, while maintaining or enhancing quality. These innovations include double courses offered during the same semester, independent study opportunities during the summer, and "late start" courses.

Many of the courses in the CIS majors form a series of prerequisites and must be taken in order. For example, the calculus perquisites must be taken before the first introductory course(113), which must be taken before 114; which serves as a pre-requisite to more advanced courses. Students can lose many semesters trying to get through this rigid sequence of courses, one at a time, meanwhile being closed out of some of the courses in the sequence some semesters.

We have redesigned three sets of courses in this series of pre-requisites as integrated "double courses" in the same semester. Students will be able to register, for instance, for 113 and 114 in the same semester. They cover 113 at twice the normal pace, devoting twice the normal number of hours to this task. By the middle of the semester, some of these "fast track" students may decide that they need a full semester after all; they will drop into a slower paced group online. Those who can learn at the accelerated pace will complete the first course in the first half of the semester, and then go on to take the follow on course (e.g.,

114) during the second half of the semester. The participating faculty members coordinate their efforts to provide a "seamless transition" between the two courses. We expect that many students will be able to learn at the increased pace because of the intensive interaction and the support available online. Every time students successfully complete a double course, they can conceivably cut an entire semester off of the number of semesters needed to finish their degree program.

Evaluation

We are using a "multi-method" approach, collecting both subjective and behavioral data, and employing both quantitative and qualitative analysis, including:

* Subjective assessments by faculty and students (based on questionnaires, interviews, and case reports by faculty). Both students and faculty must perceive benefits to this media mix in order for them to continue to select it and thus for the project to be a success in the long run.
* Quantitative/behavioral measures. These include:
    * System use: To be gathered by system monitor and automated statistical analysis routines
    * Time to Degree: We will establish baseline measures of the time currently needed to complete degree programs, by entering freshmen and by transfers full and part time, in the CIS degree programs. Four, five and six years after the project is implemented, we will examine and report data on semesters from first full time enrollment to graduation for freshmen, correlating this with number of Video plus Virtual Classroom sections of courses enrolled in.
    * Access for disadvantaged students: We will examine characteristics of students who opt to take various numbers of VC plus video courses, as compared to students registered in traditional sections, to determine if a higher proportion of women, disabled, and minority or otherwise disadvantaged students enroll in the experimental delivery mode courses.
    * Quality measures: We will compare course grades for students in experimental sections with the distribution of grades for the same course offered by the same instructor via other media.
    * Repeat customers: We will examine the proportion of first-time enrollees in Video & VC courses, who are not graduating seniors and who choose to take a second or/or subsequent courses via this media mix, as a behavioral measure of satisfaction.

Looking to the Future

When most students have multi-media work stations, the awkward logistics of tape distribution or student recording of broadcasts will probably be supplanted with CD-ROM based digitized video modules. However, for the short term, the VC + video mix seems to work well for a wide variety of courses.

A development of great import is the tying together of the world's computerized information resources, and their accessibility to students and instructors through such networks as Internet, which links most colleges and universities. This means that students can access the equivalent of the world's libraries of information in computerized form, in doing research for papers or projects. In order to make the wealth of information navigable, it is being organized in many cases into a "Hypertext" format. An example of such a system is the "World Wide Web," which is now incorporated as part of the facilities of the Virtual Classroom on EIES 2. This provides access to the Internet GOPHER world wide information services, including the ability to search databases on many campuses connected to the Internet.

The future of this technology is tied to overcoming some of the difficulties related to the current situation of budget cutting and increased course loads for faculty in higher education. The first difficulty is the initial burden placed upon instructors to completely rethink the nature of their courses and adapt their teaching to a facilitative role. It is also necessary to provide some training for faculty on how to utilize collaborative learning approaches. There is also an initial workload in terms of creating materials in electronic form that is quite large the first time one teaches utilizing this medium. Faculty may be far slower to change and adopt this technology than students; therefore, one has to consider the incentives to do so in the particular educational institution.

In the comprehensive NJIT studies of undergraduate courses (Hiltz, 1994), one key generalization was that good students often do much better in the Virtual Classroom, while poor students may do worse. As a result, the average performance is usually the same as face-to-face classes. For graduate students, there is usually a much smaller proportion of students who do not have good study habits and the self-discipline to do their studies without face-to-face supervision. In a distance education situation it is quite easy for students to put off their participation to a point where it becomes impossible to catch up with the class. The instructor must be provided, in the computer environment, with the tools to track the progress of the students and detect problems early on.

There is no comparison with other modes of delivery of distance education in terms of the ability of CMC technology to allow students to work together in groups and to become a student body with respect to the program of study as a whole. Those of us who hold that this peer group reinforcement is very necessary to creating an outstanding educational process see no other cost effective alternative to computer based conferencing. It is possible with this technology, not just to provide courses, but to establish a learning community and a "virtual university" with all the facilities necessary to make that a reality.

Acknowledgments

Development and research on the B.A.I.S. degree via a combination of video and VC is supported by the Alfred P. Sloan Foundation. Continuing research on appropriate software structures for collaborative learning via computer is partially supported by a grant from the National Science Foundation (NSF-IRI-9015236).

References

Bruffee, K. A. (1984). Background and history to collaborative learning in American colleges. *College English*, 46 (7), 635-652.

Harasim, L. Ed. (1990). *Online education: Perspectives on a new medium*. New York: Prager/Greenwood.

Harasim, L., Hiltz, S. R., Teles, L., & Turoff, M. (1994). *Learning Networks: A Field Guide*. Cambridge MA: MIT Press (in press).

Hiltz, S. R. (1986). The virtual classroom: Using computer-mediated communication for university teaching. *Journal of Communication*, 36 (2), 95-104.

Hiltz, S.R. (1993). Correlates of Learning in a Virtual Classroom. *Int. J. Man-Machine Studies*, 39, 71-98.

Hiltz, S. R. (1994). *The Virtual Classroom: Learning Without Limits via Computer Networks*. Norwood NJ: Ablex Publishing Corp., Human-computer Interaction Series.

Johnson, David W. (January, 1981). Student-student interaction: The neglected variable in education. *Educational Research* (pp. 5-10).

Johnson, D. W., & Johnson, R. T. (1975). *Learning together and alone: Cooperation, competition, and individualization*. Englewood Cliffs, NJ: Prentice Hall.

Mason, R., & Kaye, A. (1989). *Mindweave: Communication, computers and distance education*. Oxford: Pergamon Press.

McCreary, E. K., & Van Duran, J. (1987). Educational applications of computer conferencing. *Canadian Journal of Educational Communication*, 16, (2), 107-115.

Paulsen, M. F., & Rekkedal, T. (1990). *The Electronic College: Selected Articles from the EKKO Project*. Bekkestua, Norway: NKI Forlaget.

Rice, R. E. & Case, D. (1983). Electronic message systems in the university: A description of use and utility. *Journal of Communication*, 33, 131-152.

Turoff, Murray, "Computer Mediated Communication Requirements for Group Support," *Journal of Organizational Computing*, Volume 1, Number 1, pp. 85-113, 1991.

Weedman, J. (1991). Task and non-task functions of a computer conference used in professional education: A measure of flexibility. *International Journal of Man-Machine Studies*, 34, 303-318.

Welsch, L. A. (1982). Using electronic mail as a teaching tool. *Communications of the ACM*, 25 (2), 105-108.

Communicating with Virtual Humans

NADIA MAGNENAT THALMANN
MIRALab, University of Geneva
24, rue du Général-Dufour,
CH 1204 Geneva, Switzerland
E-Mail: thalmann@cui.unige.ch

Abstract: In this paper, we present an open hybrid system for facial animation. It encapsulates a considerable amount of information regarding facial models, movements, expressions, emotions and speech. The complex description of facial animation can be handled better by assigning multiple input accessories. These input accessories may be a simple script or a multi-input musical keyboard or a gesture dialogue from the DataGlove or some other type of interactive physical or virtual device. Integration of all means of control offers flexibility and freedom to the animator. The scope of such an open system is tremendous. Virtual worlds would be desolate indeed without things like synthetic faces that we can relate to and understand.

Introduction

In the context of real-time multimedia animation systems, the relationship between a real person (the user) and virtual humans should be emphasized. With the existence of graphics workstations able to display complex scenes containing several thousands polygons at interactive speed, and with the advent of such new interactive devices as the Spaceball, EyePhone, and DataGlove, it is possible to create computer-generated characters based on a full 3D interaction metaphor in which the specifications of deformations or motion are given in real-time. True interaction between the animator and the actor requires a two-way communication: not only may the animator interact to give commands to the actor but the actor is also able to answer him. Finally, we may aspire to a virtual reality where synthetic actors participate fully: real dialog between the animator and the actor. The animator may now enter in the synthetic world that he/she has created, admire it, modify it and truly perceive it. Finally, computer-generated human beings should be present and active in the synthetic world. They should be the synthetic actors playing their unique role in the theater representing the scene to be simulated.

The face is a small part of a human, but it plays an essential role in communication. People look at faces for clues to emotions or even to read lips. It is a particular challenge to imitate these few details. In this paper, we present an open hybrid system for facial animation. It encapsulates a considerable amount of information regarding facial models, movements, expressions, emotions and speech. The complex description of facial animation can be handled better by assigning multiple input accessories. These input accessories may be a simple script or a multi-input musical keyboard or a gesture dialogue from the DataGlove or some other type of interactive physical or virtual device. Integration of all means of control offers flexibility and freedom to the animator.

Facial Animation

Because the human face plays the most important role for identification and communication, realistic construction and animation of the face is of immense interest in the research of human animation. The ultimate goal of this research would be to model exactly the human facial anatomy and movements to satisfy both structural and functional aspects. However, this involves the concurrent solution of many problems. The human face is a very irregular structure, which varies from person to person. The problem is further compounded with its interior details such as muscles, bones and tissues, and the motion which involves complex interactions and deformations of different facial features. Although all movements may be rendered by muscles, the direct use of a muscle-based model is very difficult. The complexity of the model and our poor knowledge of anatomy makes the results somewhat unpredictable. This suggests that more abstract entities should be defined in order to create a system that can be easily manipulated. A multi-layered approach (Kalra et al. 1991) is convenient for this. In order to manipulate abstract entities like our representation of the human face (phonemes, words, expressions, emotions), we propose to decompose the problem into several layers. The high level layers are the most abstract and specify "what to do", the low level layers describe "how to do". Each level is seen as an independent layer with its own input and output.

There are presently three main types of facial animation systems in terms of driving mechanism or animation control. One type of systems uses a script or command language for specifying the animation (Kalra et al. 1991; Kaneko et al. 1992; Magnenat Thalmann et al. 1988; Pelachaud et al. 1991). These systems are simple but non-interactive and thus are not very appropriate for real time animation. In addition, fine-tuning an animation is difficult when merely editing the script, as there exists a non-trivial relation between textual description and animation results. Recently several authors have proposed new facial animation techniques which are based on the information derived from human performances. The information extracted is used for controlling the facial animation. These performance driven techniques provide a very realistic rendering and motion of the face. Williams (1990) used a texture map based technique with points on the surface of the real face. Mase and Pentland (1990) apply optical flow and principal direction analysis for lip reading. Terzopoulos and Waters (1991) reported on techniques for estimating face muscle contraction parameters from video sequences. Kurihara and Arai (1991) introduced a new transformation method for modeling and animating the face using photographs of an individual face. Waters and Terzopoulos (1991) modeled and animated faces using scanned data obtained from a radial laser scanner. Saji et al. (1992) introduced a new method called "Lighting Switch Photometry" to extract 3D shapes from the moving human face. Kato et al (1992) use isodensity maps for the description and the synthesis of facial expressions. However, these techniques do not process the information extraction in real-time. Real-time facial animation driven by an interactive input device was reported by DeGraf (1989), but the external control on animation is very limited when used in isolation. Though it provides high accuracy for timings, it is extremely difficult to edit. Systems driven by speech (Lewis 1992; Hill et al. 1988) are focused in lip-synchronization and speech decomposition into phonemes. These are adequate when animation involves only speech. What would in fact be more desirable, is a system which can encapsulate different kinds of animation specifications and control mechanisms. Such a system would meet the needs of the animator for almost every situation by giving access to the different means of control. The tracking of a live video sequence may provide the basic sequence of a synthetic animation, textual data may produce speech with audio feedback, a hand gesture may govern the gesture motion of the head and eyes, and so on. Here, our attempt is to present how the information from different sources can be related and controlled to give a sequence of animation. As there does not exist what one can refer to as the 'best' framework for motion control for facial animation, this suggests having an open system where one can try several possibilities and chose the one which is subjectively the 'best.'

In order to gain flexibility and modularity in the execution of the system we need a high degree of interaction. We present some of the advanced input accessories which provide natural interaction and thus intuitive control. 3D interaction is already quite popular for many applications, and here we integrate some of the novel paradigms to experiment in the context of controlling facial animation. Possibilities for control with different interactive situations are examined; e.g. gesture dialogue using a DataGlove and musical streams from a MIDI-keyboard. We believe that it is more important to provide a wide range of interaction components than to enforce a particular style of interface. One of the interactive systems for facial expressions presented by deGraf (1989) contains the philosophy of using various puppet interfaces to drive facial animation, however, it seems to have hard wiring of devices for manipulations, which restricts flexibility and interchangeability of different device components. Figure 1 shows an example of facial expression.

A Multimedia Architecture For Facial Expressions

Facial movements rely on perception-driven behaviors. Cognitively, these can be understood as externalization or manifestation of verbal or non-verbal communication agents on a face. These agents activate certain channels of a face associatively which in turn triggers the relevant muscles and which eventually deforms the face. In a computational model, such a behavior can be interpreted as translating behavioral or cerebral activity into a set of functional units which embody the necessary activity-information. The resulting actions are then combined in a sequence of discrete actions which when applied cause the necessary movements on the face. In our system we model such a behavior by separating facial animation into three major components, namely face model, animation controls and composer. The face model primarily describes the geometric structure of the face, deformation controller and muscle actions. The model receives streams of actions to perform. These actions are decomposed into the required muscle actions and a new instance of the face is derived for each frame.

Fig.1. Facial expression

The animation controls specify animation characteristics (Magnenat Thalmann and Thalmann 1991). A facial animation system needs to incorporate adequate knowledge about its static and dynamic environments to enable animators to control its execution with (maybe) predefined, yet flexible set of commands. The system's structure therefore should embed such a know-how in a natural way. In order to satisfy this need, our system employs hierarchical structure and modular design.

Animation Control and Input Accessories

As the system considers the specification of different input components separately from the animation, we can try various ways of controlling the animation. Also, various input components can be used at the same time. Such an approach provides a platform where we can experiment with new methods of control. This analysis may allow us to identify the kind of access we may require for defining and controlling varying levels of abstractions for computational animation models in virtual environments. These accessing components may demand different kinds of interaction which may establish the need for experimentation with several types of devices. As no single mode of control can give completeness, such a test-bed environment can evaluate what device can be used for which means of control. Possibility of composing and mixing different types of controls enhances the reconfigurability of the system. This also allows cooperative group tasks for animation, where more than one person can control the animation in real time. We present here four types of input accessories we have tried. All have some advantages and disadvantages. As there could be simultaneous use of different types at the same type we can overcome some of the disadvantages.

Script-based animation

Script is a standard method of specifying animation consisting of different types of entities. It is like a special language using a few key words for specific operations. Most automated facial animation systems employ this approach.

In our system a language HLSS (High Level Script Scheduler) (Kalra et al. 1991) is used to specify the synchronization in terms of an action and its duration. From action dependence, the starting and the ending times of an action can be deduced. The general format of specifying an action is as follows: while <duration> do <action>. The duration of an action can be a default duration, a relative percentage of the default duration, an absolute duration in seconds, or deduced from other actions preceding or succeeding the present action. The starting time of each action can be specified in different ways, for example, sequentially or parallel using the normal concepts of "fork" and "end" employed in scheduling problems.

One of the major advantages of such an input accessory is that it is in text form. Users can very conveniently change it by editing a text file. On the other hand, being non-interactive it is not possible to change certain parameters while the script is running. Therefore, it is not very suitable for real time animation. It is more useful for background animation.

Animation control through MIDI-Keyboard

MIDI-keyboards can be another type of source for controlling the animation. The keyboard has a number of keys enabling us to associate several parameters with the keys. Activation of each key gives two kinds of information: initial velocity with which the key is hit and the pressure variation. System G^6 a real-time, video animation system uses Korg M1 keyboard to move different parts of the mask of a face. However, the system has special purpose processing hardware to perform the animation.

In our system, this device can be used as direct manipulator for MPAs, each key may be assigned to an MPA and the initial velocity of the key may be attached to its intensity. This type of control is at a rather low level. Higher level control can also be obtained by assigning the keys to expressions and phonemes. Fig 5 shows a sequence of facial animation using the key presses of the MIDI-keyboard. Here, some of the expressions and phonemes have been associated with particular keys of the keyboard, the intensity of the expression or phoneme is governed by the velocity of the key hit. The duration of an expression can be determined by the duration for which the key is pressed. That means an expression starts with the intensity corresponding to the velocity of the key hit and continues until the key is released. At present, the pressure variation of the key hit is not being used, however, we intend to include it to modulate the intensity of an expression during its execution. Sound output from MIDI can provide the needful feedback at the execution of an action. At expression level, it may reflect its intensity. It may be interesting to use array of keys to control a single emotion, here, the mapping of keys would be with the included expression instances or channels, for example, an emotion containing eye, head and mouth motion, each may be associated with a set of keys respectively.

The advantage of such a device is that it provides a number of keys which can be assigned individually to the motion parameters. This gives simultaneous control on many parameters. But at the same time it demands hardwiring of the meaning of desirable action with a particular key. Each time a modification or an extension in the control method will require reconfiguration of the meaning of keys. Also, one-dimensional arrangement of keys in the keyboard forces a certain type of order in the manner of control. For music the ordering is with respect to frequency. However, it is not evident how to arrange facial expressions in a one-dimensional fashion, consequently this constrains the use of the device in its natural form.

Postures and Gestures Dialogue

The type of control presented in the previous section is directly dependent on the physical structure of the device used: the mapping between a user's actions and animation controls is obtained by associating a meaning to the various key presses. This device dependency is a factor that limits the animator's expressiveness.

The use of devices which simply sense user's motions, and the use of adaptive pattern recognition can overcome these problems. This gives more freedom to the animator and allows the definition of the mapping between sensor measurements and interpretations to be more complex. Hand gesture recognition is a domain where these techniques can be used. The use of hand-gestures can provide non-verbal cues for a natural human-computer interaction.

In our system, we use posture recognition on data obtained from the DataGlove to obtain categorical and parametric information to drive facial animation. In the ensuing sections we present briefly the posture recognition technique, its continuous classification and application to facial animation control.

Once a posture is recognized, parametric information can be extracted from the location of hand and how it moves. This information can then be used to drive the facial animation. The type of posture as the categorical information can be associated with a type of action performed by the face. We experimented with two types of controls: direct control at the expression level and higher level control at the emotion level.

The recognition technique here is based on multi-layer perceptrons (MLPs) (Rumelhart et al. 1986), a type of artificial neural network which is potentially able to approximate any real function (Cybenko 1989).

One of the main disadvantages using gesture dialogue as a control means for facial animation is the poor precision offered by the DataGlove. This prohibits the use of this device for fine-tuning. However, for global action manipulation, this provides a natural means of nonverbal communication and interaction method. For duration control of emotion, it is better to associate it with the duration of the gesture itself so as to establish a temporal correspondence between animator's action and the action executed.

Live Video Digitizing and Human Performances

Another approach consists of recording a real human face using a video input like the Live Video Digitizer and extracting from the image the information necessary to generate similar facial expressions on a synthetic face. The problem with this approach is that the image analysis is not easy to perform in real-time. Our recognition method is based on snakes as introduced by Terzopoulos and Waters (1991). A snake is a dynamic deformable 2D contour in the x-y plane. A discrete snake is a set of nodes with time varying positions. The nodes are coupled by internal forces making the snake acting like a series of springs resisting compression and a thin wire resisting bending. To create an interactive discrete snake, nodal masses are set to zero and the expression forces are introduced into the equations of motion for dynamic node/spring system. Terzopoulos and Waters make it responsive to a force field derived from the image. They express the force field which influences the snake's shape and motion through a time-varying potential function. To compute the potential, they apply a discrete smoothing filter consisting of 4-neighbor local averaging of the pixel intensities allowed by the application of a discrete approximation.

Our approach is different from Terzopoulos-Waters approach because we need to analyze the emotion in real-time. Instead of using a filter which globally transforms the image into a planar force field, we apply the filter in the neighborhood of the nodes of the snake. We only use a snake for the mouth; the rest of the information (jaw, eyebrows, eyes) is obtained by fast image-processing techniques.

Implementation

The System is written in C with the interface built on top of the Fifth Dimension Toolkit (Turner et al. 1990). The various input components are independent processes running on UNIX workstations. The communication between the processes is done through sockets in stream mode (sequenced two-way communication based on byte streams) using the Internet protocol. Figure 1 shows the components of a proposed system where each component is a different 3D device or a media. The input components communicate with the central process through inter process communication (IPC). A command IPC-server starts the server on the machine, the command is executed. The server basically accepts and distributes messages to all its clients. The output of each component is further processed by IPC-filter to get the desired mapping between the IPC outgoing messages of each components to the incoming action streams to the central process. The central process basically composes the IPC messages coming from different sources and produces the relevant stream of actions (MPAs) to be performed by the face model. Sensory feedback can be provided to users by different means, e.g. real time animation serves as the visual feedback, MIDI output may provide audio feedback and a text output may provide the final animation sequence script. IPC protocol allows the flexibility of developing each component individually and then hook up with the main process.

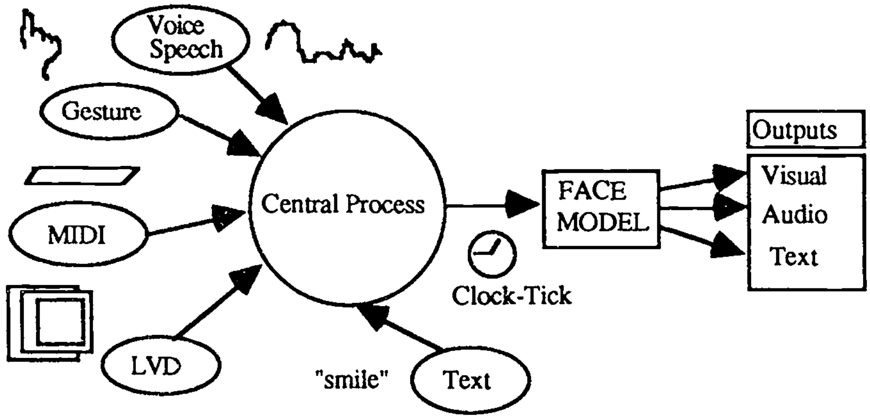


Fig. 2. Proposed system with multimedia components for facial animation control.

References

Cybenko G (1989) Approximation by Superposition of a Signoidal Function, Math, Control Signals and Systems (2), pp. 303-314.

deGraf B (1989) in State of the Art in Facial Animation, SIGGRAPH '89 Course Notes No. 26, pp. 10-20.

Hill DR, Pearce A, Wyvill B (1988), Animating Speech: An Automated Approach Using Speech Synthesized by Rules, The Visual Computer, Vol. 3, No. 5, pp. 277-289.

Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1991) SMILE: a Multilayered Facial Animation System, Proc. IFIP Conf. on Graphics Modeling, Tokyo, Japan, pp.189-198.

Kaneko M, Koike A, Hatori Y (1992) Automatic Synthesis of Moving Facial Images with Expression and Mouth Shape Controlled by Text, Proc CGI '92, Tokyo (Ed T L Kunii), pp. 57-75.

Kato M, So I, Hishinuma Y, Nakamura O, Minami T (1992) Description and Synthesis of Facial Expression based on Isodensity Maps, in: Tosiyasu L (ed): Visual Computing, Springer-Verlag, Tokyo, pp.39-56.

Kurihara T, Arai K (1991), A Transformation Method for Modeling and Animation of the Human Face from Photographs, Proc. Computer Animation '91 Geneva, Switzerland, Springer-Verlag, Tokyo, pp. 45-57.

Lewis JP (1992), Automated Lipsynch: Background and Techniques, The Journal of Visualization and Computer Animation, Vol. 2, No. 4, pp. 118-122,.

Magnenat Thalmann N, Primeau E, Thalmann D (1988), Abstract Muscle Action Procedures for Human Face Animation, The Visual Computer, Vol. 3, No. 5, pp. 290-297.

Magnenat Thalmann N, Thalmann D (1991) Complex Models for Animating Synthetic Actors, IEEE Computer Graphics and Applications, Vol.11, No5, pp.32-44.

Mase K, Pentland A (1990) Automatic Lipreading by Computer, Trans. Inst. Elec. Info. and Comm. Eng.,vol. J73-D-II, No. 6, pp. 796-803.

Pelachaud C, Badler NI, Steadman M (1991), Linguistic Issues in Facial Animation, Proc. Computer Animation '91, (Eds Magnenat-Thalmann N and Thalmann D), Springer, Tokyo, pp. 15-30

Rumelhart D. E., Hinton G. E., Williams R. J. (1986), Learning Internal Representations by Error Propagation. In Rumelhart D. E., McClellend J. L. (eds.), Parallel Distributed Processing, Vol. 1, pp.318-362.

Saji H, Hioki H, Shinagawa Y, Yoshida K, Kunii TL (1992) Extraction of 3D Shapes from the Moving Human face Using Lighting Switch Photometry, in Magnenat Thalmann N , Thalmann D (eds) Creating and Animating the Virtual World, Springer-Verlag Tokyo, pp. 69-86.

Terzopoulos D, Waters K (1991) Techniques for Realistic Facial Modeling and Animation, Proc. Computer Animation '91,(Eds Magnenat-Thalmann N and Thalmann D), Springer, Tokyo,, pp. 59-74.

Turner R, Gobbetti E, Balaguer F, Mangili A, Thalmann D, Magnenat-Thalmann N (1990), An Object-Oriented Methodology Using Dynamic Variables for Animation and Scientific Visualization, Proc CGI 90, Singapore, (Eds Chua TS, Kunii TL), pp. 317-328.

Williams L (1990), Performance Driven Facial Animation, Proc SIGGRAPH '90, Computer Graphics, Vol. 24, No. 3, pp. 235-242.

Acknowledgment

The research is supported by Le Fonds National pour la Recherche Scientifique.

Issues and Obstacles with Multimedia Authoring

Fillia Makedon
Samuel A. Rebelsky, Matthew Cheyney
Charles Owen and Peter Gloor*

Computer Science
The Dartmouth Experimental Visualizaton (DEV) Laboratory
The Dartmouth Institute for Advanced Graduate Studies (DAGS) in Parallel Computation
Dartmouth College
6211 Sudikoff Laboratories, Hanover, NH 03755
email: makedon@cs.dartmouth.edu *UBS, Union Bank of Switzerland

1 Introduction

Unlike traditional authoring, multimedia authoring involves making hard choices, forecasting technological evolution and adapting to software and hardware technology changes. It is, perhaps, an unstable field of endeavor for an academic to be in. Yet, it is important that academics are, in fact, part of this process. This paper discusses some of the common threads shared by three dissimilar cases of multimedia authoring which we have experimented with, that of multimedia conference proceedings, multimedia courseware development and multimedia information kiosks. We consider these applications from an academic point of view and review the benefits and pitfalls of academic development while sharing points of hard-learned wisdom. We draw on experiences from some of the projects run at the Dartmouth Experimental Visualization Laboratory (DEVlab), where we have been developing different types of multimedia applications.

A multimedia document (or application) can be anything from an electronic book to an interactive course, from an interactive slide presentation to a multimedia newspaper or an orientation tool, from an interactive auto manual to a clinical record. Multimedia authoring systems come in many forms, depending on the different applications that drive them. Applications range widely and include: office documents, conference proceedings, information kiosks, professional brochures, course materials, virtual reality museum presentations and others yet to be discovered.

1.1 Multimedia Systems and Tools

The greatest challenge in using multimedia authoring tools is that of automating the process of integration and organization. The effective integration of software programs that enable a user to "experience" a concept through several modes and media (such as text, audio, video, simulation, graphics, visualization, animation, slides, pictures, images, etc.), is still a state of art, rather than science. Multimedia authoring has not reached a stage where we can prove that learning, for example, is more effective than with traditional teaching methods.

Just as multimedia authoring systems vary, so do the users of these systems. New applications lead to the emergence of new users who can range from graphic artists to computer science graduate students learning computational biology principles, or from healthcare professionals being trained on a new machine to history students preparing presentations. One desirable feature behind all these applications is interactivity. Interactivity places the user (or learner) in control of the system; the user manipulates the media forms present through different possible modes of interaction. Interactivity may also mean that it is possible for multiple authors and or users to collaborate.

An easy classification of multimedia authoring systems can be based on three primary platforms: (1) the IBM PC compatibles (DOS, Microsoft Windows or OS/2), (2) the Macintosh, and (3) the UNIX workstation (SUN, DEC, SGI). In addition, the Power PC promises to provide an appropriate platform once it is released and gains acceptance. For an educator, the widest audience can be found in the IBM

PC platform today. In 1991, the Multimedia Personal Computer (MPC) consortium defined a minimum specification for an MPC. On the other hand, the Apple Macintosh, with its QuickTime capabilities has offered exceptional ways of integrating time-based media. UNIX workstations have the problem of dissemination to overcome still, since not all end-users own a SUN, or a DEC or an SGI.

Because of local expertise and resources, our projects at Dartmouth have been primarily designed on the Macintosh. In order to make these projects available on other platforms (typically a PC), we redevelop using related tools (e.g., ToolBook on the PC as a replacement for HyperCard on the Macintosh). Keeping in mind that disseminations is one of the main bottlenecks in making a system highly usable, we have also adopted CD-ROM technology. CD-ROM technology provides large random access storage at low cost, which makes possible the "fancy," typically space consuming, features such as audio, motion video, and interactive animations, which drive the market for such products. Furthermore, CD-ROM devices are relatively cheap, almost absolutely conform to a single standard, and are highly popular, especially in academic circles. The reason for this popularity is portability. A CD represents a mini digital library in one's pocket. A CD also consumes little of a user's limited disk resources, which is quite important in the academic world.

2 Steps in the Production of Multimedia Documents

Before we proceed with describing three cases of multimedia authoring, we take a brief look at the tasks involved in a multimedia production effort. We can divide them as follows:

(a) *Requirements definition:* This task involves assessment (cost, time framework, resources available), definition of user profile, prioritization of desirable facilities included in the application and based on the preceding two factors, evaluation of dissemination platform choices and estimation of the evaluation, production and dissemination costs. This task also involves market analysis (rarely done in academic circles -- for example, how easy is it to sell a proceedings in parallel computation in a bookstore that does not have the facilities to demo it?).

(b) *Software tool search.* The second task is that of researching a wide range of software that matches with the platforms chosen in (a), and identifying all the specialized tools to integrate. The choice of a particular authoring system (or configuration of authoring systems and translation tools) for a particular application is critical to success. This is a complex and very time-consuming process that can evolve during production. For example, one may compromise and trade off one feature for another due to the changes in the constraints of an application or other technological changes.

(c) *Content research, media orchestration and design integration phase.* The material must be researched, organized, assimilated, written, and a script produced which, like a theater play, orchestrates the appearance and activation of various components and media at designated times. Since, in a hypermedia environment, the user may have alternate options of exploration, this must be taken into account. This stage involves a cyclical process of editing, evaluating, editing, expanding, evaluating, etc. The content (domain) and the users of this content are two driving forces that determine the type of user interface, the trade-off between its sophistication and ease of use, simplicity and power. For example, the built-in support facilities for sound and video or for creating interactive animations, are major factors in making the system more usable and yet more complex.

(d) *Programming, integration, evaluation, and stepwise refinement.* The programming of the application is done in a "build a little, test a little" mode. This task involves putting a group of programmers together (mostly students), training them, coordinating them and supervising the progress. This process is problematic in most academic environments, due to low cost student programming which takes second place to course work, etc. It is important to also have available a separate focus group of users (faculty/students) who evaluate individual components of the product and provide feedback during production.

(e) *Product Distribution and update maintenance of the software.* This task usually falls into the hands of the production manager (usually a faculty) and the publisher.

3 Case 1: Interactive Conference Proceedings

So, the question is, "*Why use/create multimedia proceedings?*" The first reason is that, instead of a volume of scientific papers, a simulation of the actual conference is beneficial in giving additional insight into the complex topics presented in a short scientific presentation at a conference. Preserving the presentation is (a) educational in seeing how the written material is encapsulated in a short talk; (b) educational in teaching how to give a scientific presentation; (c) a cost effective way of attending a conference (or having your students do) without leaving home. In fact, multimedia proceedings can serve as the new way of interviewing candidates for academic positions, or simply getting to know who are the key researchers in a field and how they relate the materials in their paper to their previous results.

Another important reason is interactivity: one can insert facilities for annotation,placing bookmarks, searching for a reference, being able to print a whole paper without copyingpage by page the conference proceedings, being able to synthesize new documents out of two or more papers, or even being able to create slides for a lecture. The possibilities are endless, not to mention the ability to create digital library archives out of such multimedia documents. We have incorporated many of these facilities in our interactive multimedia conference proceedings, and this process has been presented in another talk in this conference.

3.1 Experiences with Proceedings from a Parallel Computation Workshop

We have obtained the materials of our multimedia proceedings from presentations and tutorials during an annual summer institute that takes place at Dartmouth and is called DAGS, (The Dartmouth Institute for Advanced Graduate Studies in Parallel Computation). Every year, DAGS emphasizes a particular aspect of parallel computation and includes invited talks, panel discussions, as well as introductory tutorials for young researchers and presentations of new research.

The DAGS '92 proceedings were on the *Practical Implementation of Parallel Algorithms and Machines* and are published as a CD-ROM by Springer-Verlag. This CD includes both the actual papers in hypertext format and the talks in multimedia (with video and sound as well) format. The CD also provides many other features, including search mechanisms, provisions for users to annotate individual slides and sections of the paper, user-defined "paths" through the proceedings, and bookmarks that allow one to quickly return to a particular part of the proceedings. These first two features make it very easy for teachers to provide guides to the material.

What is fundamentally different about these proceedings, as opposed to, say, the CD-ROM proceedings produced for the ACM'93 multimedia conference, is the fact that we produced them after the conference was over, rather than hand them to the participants at the time. This had the advantage of the time element: we were able to re-process materials and involve the authors more actively. In fact, for DAGS'93 we plan to also include the participation of some of the organizers of the symposium as well as profiles of some of the invited speakers and the participants of the "school" that followed the symposium.

The theme of DAGS'93 was"Parallel I/O and Databases", a very important topic in the construction of efficient parallel machines. The 1993 proceedings will be quite different from DAGS'92; it will not only be for two platforms, the Mac and the PC/Windows, but also have more features that enable someone to feel as if she were at the conference itself. For example, while the slides in the DAGS '92 proceedings were mostly static, the DAGS '93 slides will include animated pointers so as to better replicate the talk. In addition to scientific talks presenting new results, the DAGS '93 proceedings will also contain tutorials on parallel programming, a separate school activity that followed the symposium. We hope that this will expand the role of the DAGS'93 proceedings as a teaching tool as well. An abstract screen dump is shown in Figure 1.

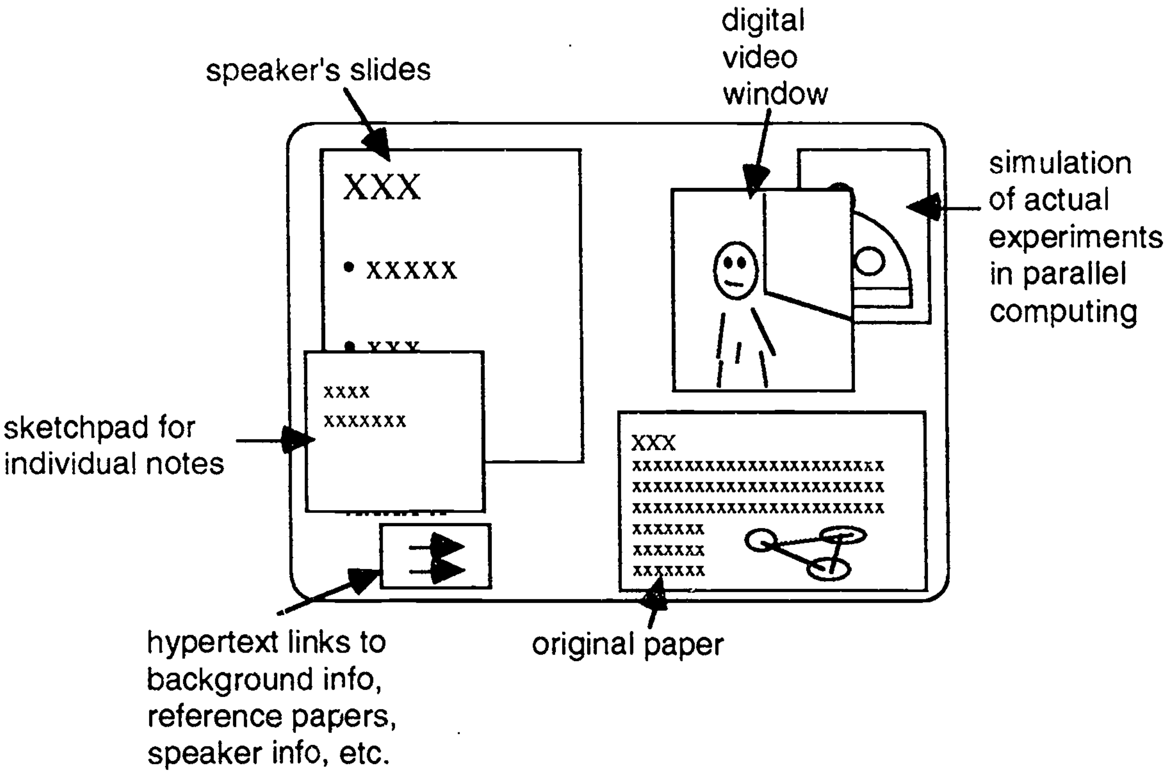


Figure 1: Example of what a screen from our multimedia proceedings

3.2 New Roles of Editors, Authors and Publishers in Multimedia Publishing

We see five primary players interacting in the publishing world: authors, editors, publishers, marketers, consumers. We will concentrate on the first three. In the world of multimedia publishing, these roles have obtained new dynamics and the relationships are continuously changing. Traditionally, the biggest task has been upon the author. A new complication with multimedia publishing is that there is not just one author but many authors, including multiple secondary-authors, i.e., content-knowledgeable programmers who implement creatively the script of the primary author (or even design the script based on the materials in a traditional text). Furthermore, we do not just have the editing of text but the need to orchestrate a whole set of media other than text, based on content. In addition to the multiple media and authors, there are options for the user to read/interact/experience the system.

In the traditional world of publishing, there isn't usually a person to manage all these processes since the publisher's role has now become that of an involved "reviewer" as well as market assessor, who comments on what needs to be changed. One cannot simply "transfer" the traditional publishing roles to the multimedia publishing arena. Here, product coordination needs a so-called production manager whose task is to coordinate the author(s), the system, the programmers, and the publisher.

In terms of editors, while in traditional publishing authors provide text and editors provide layout, formatting, and other constraints, while in multimedia publishing, editors may undertake a greater responsibility in the production, depending on technical expertise. An editor must, for example, have an understanding of what it takes to present/orchestrate the multimedia data in a correct format. In fact, both editors and authors are of a new technical breed whose communication includes content and implementation.

The role of the multimedia author is now split, depending on the level of involvement; there may be a script-type of author, working with programmer-authors or programmer-editors. Authors now face new submission requirements, even if their task is to provide a script to a programmer. They must understand the system well in order to ensure successful and timely production of a given application and they may be required, for example, to provide fully formatted text to their publishers. In producing next year's proceedings, DAGS '94 (theme: Parallel Programming Environments and Problem Solving Environments), we will give each author tools to develop their talks in such a way that we can easily add it to the proceedings without extensive work on our part. For DAGS '94, we hope to convince the authors to:

- "chunk" their text, adding links and making clear the relation between their slides and the corresponding full-text paper, and between the paper and the references
- make the relation between the material in the paper and the material in the talk explicit;
- build or describe their own links to other multimedia documents in the proceedings.

In producing conference proceedings, one needs to overcome the diversity of software tools available and the resistance on the part of authors to use certain tools to create graphs or slides for their presentation. To automate the process, it is important to be always aware of new tools that emerge which allow the translation of software into each other and the integration and interaction of software from different platforms. These tools are essential for making a system portable, extensible, and disseminable.

4 Case 2: Authoring Multimedia Courseware

Unlike conference proceedings which present front-line research issues (such as, in our case, topics in parallel computing), multimedia courseware presents a different challenge: how to compact knowledge of a possibly complex nature, and present it in an aesthetic, simple, interactive way to novices that motivates learning. We will now discuss our experiences with authoring multimedia modules for a course to teach parallel computing to freshmen and sophomores, a project funded by the National Science Foundation and the New England Consortium for Undergraduate Science Education.

In this project, one of the questions has been how to effectively harness visualization and multimedia technology in order to achieve fast and pleasant learning. Our hypothesis has been that the content (syllabus) drives the design and production of such a course. Furthermore, we believe that an important criterion in the process of integration of the various materials and media is asking the question, "which is the best way to present this concept," as opposed to, "I have this video clip of x,y and this animation of x,z: how and where do I fit them into my application?" In other words, we advocate a minimalist approach which, while it aims at presentation richness, stays focused on the content to be covered and avoids "information clutter." Our aim is to make this course available widely, to computer science faculty and students nation-wide who may have diverse equipment resources.

4.1 Teaching Parallel Computing to Novices

The course is being designed to allow the student to develop fundamentally new ways of thinking, new ways of organizing and viewing real-time data and, most importantly, new ways to learn and practice computing for parallel solutions and in parallel instructional environments. Given that parallelism is harder to comprehend and teach (concurrency and asynchrony are two of the main difficulties in visualizing parallel processes), it is quite a challenge to present such materials to novices. Nonetheless, we believe that parallelism should be at the heart of the computer science curriculum and that it is the future of computing and computer science. The successful introduction to thinking in terms of parallel execution early will result in our eventually rebuilding our curriculum and providing a national model for other schools to do the same.

This cour   ,ill be novel in two ways. The first novelty is the introduction of this material before algorithms and data structures have been completely introduced, as we currently conceive them in a sequential setting. Unlike courses which "add-on" two or three parallel laboratory components to an existing data structures course, we will concentrate on developing a course which is cohesive and guarantees unity. To our knowledge, this has not been done before. The second novelty of our course is the use of visualization with minimal "informational clutter", hypermedia-induced disorientation, or exposure to unnecessary details concerning parallel machine implementations and parallel languages. We believe that a complex web of information can overwhelm the novice student and it is best to keep each interactive module separate and centered on a particular topic and example. This allows the students to focus on the material at hand, and not lose themselves in an overwhelming array of information. It also allows us to gradually add new concepts as the course progresses (so that students don't accidentally skip to the last topic and become so confused that they give up).

Algorithm visualization plays an important role in the authoring of multimedia courseware for computer science. One problem is, however, that algorithm visualization is quite different from traditional, scientific visualization. While most fields are grounded in real objects, Computer science algorithms involve abstractions which lack concrete representation; learning how the visual objects correspond to the abstract objects of an algorithm represents an additional concept for the novice student.

Course production planning involves, therefore, a choice of what to include and how to present it effectively, in a way that motivates a parallel solution. Secondly, it involves a careful review and prototype development of a shell and a graphical user interface, which should be consistent throughout the course, easy to use, cost effective, easy to port to other environments, easy to extend, and easy to maintain. Since we have divorced the programming skills that the author-user needs to have from the skills that the end-user needs to have to experience the course, neither student nor teacher need to know the authoring programing language to use the course.

As a primarily laboratory-based course, it will involve the students in parallel programming using interactive visualization tools and integrated multimedia facilities. Multimedia tools will be used for cooperative work, student-teacher communication, homework production, course management, solution dissemination and faculty presentation. The course will be composed of eleven multimedia modules to represent ten weeks of a laboratory-based course. The first module provides a self-contained interactive guide for the user to practice with and to "preview" the remaining modules.

We have decided to de-emphasize the mechanical aspects of parallel computation, (e.g., programming implementation details, interconnection protocols, parallel performance issues such as load balancing), and put more emphasis on the intellectual (algorithmic) impact of parallel computation. Unlike courses which present topics in the form of a series of "smart" projects, we aim to produce a novel and integrated intellectual view of data structures. We achieve this by concentrating on the concepts of parallel computing, rather than on a series of parallel languages and systems that, like a black box, produce an answer fast but provide no understanding of the underlying principles. We visualize the data to show how processors are applied to data, thus focusing on the algorithms.

5 Kiosks—Interactive Multimedia Brochures

One of our initial forays into the world of multimedia authoring has been the development of an interactive guide (kiosk) to computer science at Dartmouth. This kiosk includes descriptions of the computer science courses offered at Dartmouth (text); short statements by the faculty members that describe their research interests and reasons for being at Dartmouth (audio and video); a map of Sudikoff Laboratory, our new research center (graphics); and a variety of other materials in many formats. The kiosk includes links between the various sections so that one may, for example, jump from a course to the professor teaching the course, to the map of the building with the professor's office highlighted.

The kiosk project required us, more than the other two projects, to examine the management aspects of the multimedia authoring experience. We had to decide which materials to include; hire undergraduates to work on the project; build a timetable; obtain or construct the need materials; convert them to digital format; decide upon an overall interface to provide so that novices can easily navigate through the various types of information; coordinate material collection (e.g., videotaping of the faculty) and put it all together. Note that the kiosk is not a static application: because the department does not remain stable (courses are added and removed, teaching assignments change, faculty members arrive and leave), the kiosk needs to be updated regularly. This feature is less obvious in the two previous applications described above.

Because of our success with the Computer Science kiosk, we have since been involved in the development of other kiosks, ranging from a guide to computing services at Dartmouth to an orientation tool for new employees at Dartmouth, to a counselor-kiosk for premedical students. We recently began to develop WisKit, an interactive tool to attract women and minorities to the field of Computer Science.

With the help of six first year students funded by Dartmouth's Women in Science Program (WISP) and the New England Consortium for Undergraduate Science Education (NECUSE), we are experimenting with the development of this very different kiosk: (a) the students have had no previous expertise in multimedia authoring; (b) while the information included in the other Kiosks could be described and obtained in advance of the authoring, the WisKit requires much more gradual development; we learn more about the material as we progress (in part, because the young women developing the WisKit discover and suggest it), and therefore find new things to incorporate. This project suggests a "build a little, test a little" development paradigm in multimedia publishing. A group of three senior upper classmen are assisting with the training of the first year students. and we are investigating a network interface to other kiosks within Dartmouth.

6 Commonalities and Lessons Learned

While proceedings, brochures, and teaching materials fall into quite different domains, we found that the preparation of interactive multimedia documents for each of these domains shared many problems and solutions. We employed a similar process going from basic idea to final product in each situation. All have a short expected delivery time and therefore require authoring tools that allow us to meet that schedule.

Proceedings are expected to disseminate current information and should be made available as soon after the conference is held as is possible. Delays can occur from inappropriate materials (e.g., due to faulty videotaping; back-up videotaping person is recommended), digitizing, editing, scanning, OCR, and formatting. Delays can also occur due to the features one chooses to include (for example, we felt that we should include an extensive network of links and, while, creating links does not take an inordinate amount of time, determining links does).

Another delay may occur due to unexpected editing needs. For example, we expected to be able to use the audio and scanned slides with few changes. The audio required significant editing (removing noise and interjections, amplification, and more). Although the slides were readable at full size (approximately 640x480 pixels), when reduced to a size compatible with CD bandwidth requirements (280x312 pixels), they were virtually unreadable and, in most cases, had to be partially or completely retyped. While we have tried to get around some of these problems in the DAGS '93 proceedings by requiring authors to submit materials in electronic form, we have found that many of these problems still recur even when all the files are available in electronic form because it is difficult (and often impossible) to convert from one format to another. Thus, the solution to automating the production still remains unsolved.

One major area of difficulty is the lack of standardized cross-platform tools. While we appreciate the differences between the Unix, PC, and Macintosh environments, it should be possible to develop presentations that are available in all three, or at least the two mass market personal computer environments, without a complex and manually assisted translation process. Though presentation tools exist that provide this basic capability, such a product does not currently exist in the authoring world.

Another issue of concern is the lack of sophisticated text manipulation capabilities in most multimedia software systems. Yes, motion video and sound are impressive, and powerful graphics help represent information more compactly, but both the course project and the conference proceedings require delivery of a large amount of text. The proceedings add powerful presentations to the basic backbone of the presented papers. However, we have found basic text capabilities such as the presentation of highly formatted documents, search engines, automatic keyword indexing, and cross references between documents to be sorely lacking. Hence, we have to build such tools either entirely manually or using scripting languages. This is surprising since so many multimedia presentations contain large amounts of text (encyclopedias, book and document collections).

It has been hard to evaluate the results in the different applications described. One difficulty is due to the fact that the criteria used are not standard and change with the application, the domain, and the type of user who is doing the evaluation. "Learning effectiveness" is a very hard thing to assess and it remains an open questionas to how multimedia-based teaching compares to traditional methods in this respect. The constraints of the university environment had a major impact on the way we developed these documents and our success in doing so. Since a great deal depends on available funding for the student programmers and since these projects are very time consuming, it is extremely hard for a single faculty to develop an application as described above without a large amount of faculty release time and/or student assistants. One possibility is that the university will view this development as infrastructure, rather than as writing a book, and provide facilities for linking to other such resources on campus electronically. This is already happening, to an extent, at Dartmouth.

Among the lessons we have learned is that many of our problems could have been avoided if we had begun with a better idea of what the end product should be like and what form the sources should take to ease development. Furthermore, understanding better our limitations on time and programmer resources, would have helped us make more conservative decisions. Finally, to meet deadline and quality measures, as in all projects, it is essential to keep tight control on the project and to meet with the project team regularly (preferably weekly).

Bibliography

1. *Parallel Computation: Practical Implementation of Algorithms and Machines.*, editors: J. Matthews, F. Makedon, P. Gloor, P. Metaxas and D. Johnson. Telos/Springer-Verlag, Santa Clara, CA, August 1993.
2. "VideoScheme: A Research, Authoring and Teaching Tool for Multimedia", J. Matthews, F. Makedon and P. Gloor, Proceedings of EdMedia 1994.
3. "Conference on a Disk: An Experiment in Hypermedia Publishing", Fillia Makedon, P. Metaxas, J. Matthews, P. Gloor, D. B. Johnson, Matthew Cheyney, Proc. of EdMedia '94.
4. "VideoScheme: A Programmable Video Editing System for Automation and Media Recognition", F. Makedon, J. Matthews and P. Gloor, 1993 Proc. of the ACM Multimedia Conf.,Anaheim, CA 8/1-8/6/93.
5. "Building a Digital Library with Extensible Indexing and Interface: A Discussion of Research in Progress", D.B. Johnson and F. Makedon, Digital Libraries Workshop, Rutgers Univ., 5/94.
6. "A Digital Library Proposal for Support of Multimedia in a Campus Educational Environment", Fillia Makedon and Charles Owen, EdMedia '94 Poster Presentation.

Lecturing in the Future: Bringing It All Together

H. MAURER
Graz University of Technology/Austria and University of Auckland/New Zealand
email: hmaurer@iicm.tu-graz.ac.at

For over three decades innumerable attempts have been made to support training, teaching and lecturing processes using computers. Despite the huge amounts of funding, energy and enthusiam that have been poured into such undertakings, results have often been less than expected. This is true if one looks at stand-alone CAI applications of most varieties of client-server approaches within LANs such as COSTOC described in [Huber 1989], is true of many simulation and intelligent tutoring systems, and also applies to numerous attempts employing high-tech teleteaching.

The advent of sophisticated computer based multi-media systems alone does not change this situation dramatically; however, networked variants thereof and new ways of looking at the whole multi-media communication process are starting to provide new scenarios and might well result in the long awaited breakthrough ([Maurer 1992], [Carlson 1992], [Lennon 1994a]).

In this talk, we show how modern hypermedia systems such as Hyper-G [Kappe 1993] can be used as a basic platform for effective multi-media delivery of lectures if backed up by a library of reference material ([Calude 1994], [Maurer 1994]). Expanding on ideas presented in [Lennon 1994b], we explain how a lecture based on material drawn from a fileserver, modified in real-time if necessary and multi-cast to student workstations not only provides a most satisfying teaching envrionment but that—using a number of additional features—it at the same time solves long open problems of courseware authoring, student-teacher and student-student interaction, and the use of CAI material and high-quality distance teaching. As a most surprising aspect, we claim that courseware authoring "on the fly" is starting to become almost a reality.

We contend that such systems can be built with moderate effort today. We report on the issues involved and on progress made in implementing such systems, thus demonstrating that our ideas are indeed more than "vapour-ware".

If the above sounds too good to be true you better come to the presentation so that we can try to convince you of the feasibility of all important issues involved!

References

Carlson, P., & Maurer, H. (1992). Computer Visualization, a Missing Organ and a Cyber-Equivalency. *Collegiate Microcomputers 10 (2)*, 110-116.

Calude, C., Maurer, H., & Salomaa, A. (1994). JUCS - the Journal of Universal Computer Science. *Report, Department of Computer Science, Unviersity of Auckland;* submitted for publication elsewhere.

Huber, F., Makedon, F., & Maurer, H. (1989). HyperCOSTOC - a Comprehensive Computer-Based Teaching Support System. J.MCA 12, 293-317.

Kappe, F., Maurer, H., & Sherbakov, N. (1993). Hyper-G: A Universal Hypermedia System; *Journal of Educational Multimedia and Hypermedia 2 (1)*, 39-66.

Lennon, J., & Maurer, H. (1994). Lecturing Technology: A Future with Hypermedia. Report, Department of Computer Science, University of Auckland (1993); to appear in: *J. Educational Technology* (1994).

Lennon, J., & Maurer, H. (1994). MUSLI - A MUlti-Sensory Language Interface; *Educational Multimedia and Hypermedia Annual, 1994,* Association for the Advancement of Computing in Education, Charlottesville, VA USA.

Maurer, H. (1992). Why Hypermedia systems are Important. *LNCS 602*, Springer Pub.Co., Heidelberg/New York, 1-15.

Maurer, H., Muelner, H., & Schneider, A. (1994). An Electronic Library and its Ramifications. *Report 382, IIG Graz* (1994); submitted for publication elsewhere.

Artificial Intelligence and Educational Technology: A Natural Synergy

-Extended Abstract-

GORDON I. MCCALLA

ARIES Laboratory, Department of Computational Science
University of Saskatchewan, Saskatoon, Canada S7N 0W0
E-Mail: mccalla@cs.usask.ca

Abstract: It is argued in this extended abstract that educational technology and artificial intelligence are natural partners in the development of environments to support human learning. Moreover, artificial intelligence techniques are useable now in practical systems. To illustrate this, several working systems that use artificial intelligence and that have been developed in the ARIES Laboratory are discussed.

It is sometimes assumed that the key to supporting learners is to provide them with a "flashy" interactive discovery environment that has fancy graphics, lots of colour, sound, sturm und drang. This is contrasted to "stodgy" old "computer assisted instruction" (CAI) systems that march the learner lockstep through a pre-programmed instructional sequence. I would like to argue that neither the flashy discovery environment nor the stodgy old CAI system will stimulate learning in the long term. The key to motivating a learner for the long haul is to provide him or her with a rich learning environment that

- supports the learner's development at the deep cognitive levels;
- allows the learner choice in setting goals (within the context of the domain);
- responds to the learner's changing needs;
- helps the learner to focus on a task or goal that is particularly relevant;
- knows the domain and ways of learning the domain;
- communicates with the learner in ways natural to the learner.

The essential point is to combine learner control with deep support. The learner is neither left on his or her own, as in the flashy discovery environments, nor dictated to as in the stodgy CAI approaches.

Designing systems with these characteristics is the long term goal of research in intelligent tutoring systems (ITS). Unfortunately, building each of these characteristics into a system is extremely difficult. Each requires the use of techniques from artificial intelligence (AI), including knowledge representation (if the system is to "know" the domain), diagnosis and user modelling (if it is to "understand" learner behaviour), planning (if it is to help the learner focus on relevant goals or tasks), machine learning (if it is to understand various ways of learning about the domain), and natural language understanding (if the system is to communicate with the learner on his or her own terms).

There have been two main criticisms of the ITS agenda:

- ITS is following the wrong star; it is trying to replicate the human teacher, rather than using computing technology in ways more natural to the technology;
- the AI techniques upon which ITS depends are not well enough developed to be used in actual systems.

Neither criticism is fully justified. There is nothing explicitly in the ITS agenda that necessitates replicating the functions of a teacher. In fact, the word "tutoring" in "intelligent tutoring systems" is meant to indicate flexible notions of supporting learning, as a tutor does, rather than directing learning, as a teacher does. Even if the goal is to enhance a discovery environment it is necessary to employ many of the same AI techniques that would be needed to replicate a teacher. Thus, the functionality and interface of a discovery environment could be tailored to meet the individual needs of a learner through user/learner modelling. The discovery environment could provide knowledge-based help and/or advice using natural language. Tasks to work on could be suggested based on an instructional plan being kept within the discovery environment.

The second criticism seems more justified. Many ITSs have indeed had to drastically narrow their domain or restrict the learner in order to keep within the capabilities of the system. The AI problems are not nearly solved. However, this doesn't mean that AI can't be gainfully employed to provide "value-added" to a learning environment, even without fully solving all of the AI problems. This is particularly true if clever ways are found to offload from the learning environment the need to be fully responsible for all possible behaviour patterns of the learner. As Self (1990) discusses, such offloading can either be on to the learner or onto a human facilitator working in symbiosis with the learning environment.

Much of our research in the ARIES Laboratory at the University of Saskatchewan is aimed directly at the second criticism. We have been explicitly exploring how AI techniques can be used in real systems to be deployed with actual human learners. This has required us to take a variety of approaches. We have refined and extended the AI techniques themselves, as in our work on granularity for diagnosis.(the SCENT project (McCalla, Greer, and the SCENT Research Team,1990) or our work on instructional planning (carried out in collaboration with Darwyn Peachey and Barbara Wasson and reported on in (McCalla, 1992)). We have been able to make clever use of existing AI techniques (as in the VCR Tutor (Mark and Greer, 1991)). We have carried out basic research into AI issues (as in our work on belief revision for student modelling (Huang, McCalla, Greer, and Neufeld, 1991)). We have been able to find applications suited to a particular AI technique (as in our learning by teaching system that exercises machine learning techniques (Srinivas, Greer, and McCalla, 1991)). It has even proven possible to take advantage of the social context of a particular learning situation in order to create a robust learning environment (as in the G.E.N.I.U.S. program advising system (McCalla and Murtagh, 1991)).

In this talk I will discuss several of these systems, showing how AI can be practically employed now to enhance the capabilities of real systems for real world use. The systems I will present have all been implemented and tested with human subjects.

In particular, the SCENT advisor will be discussed, and I will show how the granularity-based representations (McCalla, Greer, Barrie, Pospisil, 1992) we have developed in the ARIES Laboratory can be used to provide robust diagnosis in a wide variety of problem solving domains. SCENT also comes equipped with the AROMA knowledge engineering environment which makes the development of a new application much easier than it would be otherwise. SCENT has been tested in the domain of LISP recursion in a couple of studies, one of which involved linking the system to another ARIES system, the scaffolding environment PETAL (Bhuiyan, Greer, and McCalla, 1992), in order to provide a distance learning capability. This study is reported on elsewhere in this conference.

I will also present the learning by teaching system (Srinivas, Greer, and McCalla, 1991). In learning by teaching the system inverts the usual instructional paradigm. The system acts as an inquisitive learner, thus stimulating the human learner to refine and extend his or her knowledge. Learning by teaching is still a prototype system, but does make clever use of machine learning techniques from AI in a way that does not require these techniques to have cognitive fidelity. Moreover, the system is very robust, working in any fact-based or taxonomic domain without needing any *a priori* specialized tuning.

The next system, G.E.N.I.U.S. (McCalla and Murtagh, 1991), takes advantage of the credibility invested in a programming advisor by human learners in order to provide "ignorance-based" advice on programming errors. G.E.N.I.U.S. uses a discrimination net to provide standard stimulus-response answers to questions from the learner. The result is a remarkably robust system that is relatively easy to engineer, although in its present form only partially successful in its effect on learners.

Finally, the VCR Tutor (Mark and Greer, 1991) will be presented. The VCR Tutor provides help to learners on how to program a video cassette recorder. But, it's main goal has been to show that a knowledge-based approach actually enhances learning. To this end four different versions of the VCR Tutor, ranging from a knowledge-based version to an environment that merely provides opportunities for exploration without feedback, were tested with 20 subjects each. This large-scale empirical study showed a clear positive relationship between the knowledge-based approaches and the capabilities of the human subjects in programming the VCR, as demonstrated in a post test. This study shows that AI indeed can provide value-added to a learning environment.

The general lesson of this talk is that far from being diverse research endeavours, AI and educational technology can interact in a natural synergy to the mutual benefit of both.

Acknowledgements

I would like to thank all of my colleagues, students, and research assistants over the years who have been instrumental in the development of the ideas in this paper. I owe a particular debt to Jim Greer. I would also like to acknowledge the financial support of Canada's Natural Sciences and Engineering Research Council and the IRIS Networks of Centres of Excellence project.

References

Bhuiyan, S., Greer, J.E., and McCalla, G.I. (1992). Learning recursion through the use of a mental model-based programming environment. *Proceedings of the 2nd International Conference on Intelligent Tutoring Systems (ITS 92)*, Montreal, Quebec, 50-57.

Frasson, C. and Gauthier, G. (eds.) (1990). Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education, Ablex, New Jersey.

Jones, M.L. and Winne, P.H. (1992). Adaptive Learning Environments: Foundations and Frontiers, NATO ASI Series F85, Springer-Verlag, Berlin-Heidelberg.

Huang, X., McCalla, G.I., Greer, J.E. and Neufeld, E. (1991). Revising deductive knowledge and stereotypical knowledge in a student model, *International Journal of User Modelling and User Adapted Interaction, 1*, 1, pp. 87-115.

Mark, M. and Greer, J.E. (1991). The VCR Tutor: Evaluating instructional effectiveness. *Proceedings of the 13th Annual Meeting of the Cognitive Science Society*, Chicago, 564-569.

McCalla, G.I. (1992). The search for adaptability, flexibility, and individualization: Approaches to curriculum in intelligent tutoring systems. In Jones and Winne (1992), 91-122.

McCalla, G.I., Greer, J.E., Barrie, J.B., and Pospisil, P. (1992). Granularity hierarchies. *International Journal of Computers and Mathematics, Special Issue on Semantic Networks, 23*, 2-5, 363-376.

McCalla, G.I., Greer, J.E., and The SCENT Research Team (1990). SCENT-3: An architecture for intelligent advising in problem solving domains. In Frasson and Gauthier (1990), 140-161.

McCalla, G.I. and Murtagh, K. (1991). G.E.N.I.U.S.: An experiment in ignorance-based automated program advising. *AISB Newsletter, 75*, United Kingdom, 13-20.

Srinivas, P., Greer, J. E. and McCalla, G. I. (1991). Learning by teaching. *Proceedings of the International Conference on the Learning Sciences*, Northwestern University, 357-363.

Self, J.A. (1990). Bypassing the intractable problem of student modelling. In: Frasson and Gauthier (1990), 107-123.

Regarding the *I* in ITS: Student Modeling

VALERIE J. SHUTE
Armstrong Laboratory
Brooks Air Force Base, Texas 78235-5352 USA
shute@eis.brooks.af.mil

Abstract: For an intelligent tutoring system (ITS) to earn its "I", it must be able to (a) accurately diagnose students' knowledge structures, skills, and/or learning styles using principles, rather than pre-programmed responses, to decide what to do next, and then (b) adapt instruction accordingly. While some maintain that remediation actually comprises the "T" in intelligent tutoring systems, my position is that the two components (diagnosis and remediation), working in concert, make up the intelligence in an ITS. A framework for developing and assessing student models is presented, followed by a description of an attempt to apply the framework in the development of a student model incorporated within a non-intelligent computer tutor. The two systems (with and without a student model) will be compared in terms of learning outcome and efficiency measures.

Jeremy (age 10) arrives at his math lab where he sits in front of a computer that is going to help him learn to solve algebra word problems. Today's focus is on those troublesome distance-rate-time problems. After stating his name, the computer accesses Jeremy's records, flagging his relevant strengths and weaknesses (i.e., not only his higher-level aptitudes from his computerized school records, but also the low-level rules that he's acquired and not yet acquired in this module). Beginning with an animated review of concepts and skills that he learned the day before, the computer generates a problem which is just a little bit beyond his grasp. The system then works out the correct solution to the problem, along with some alternative solutions that Jeremy is likely to come up with based on its student model of him. In fact, he incorrectly solves the problem like the tutor predicted. As part of its student model of him, the computer "knows" to instruct Jeremy with an emphasis on a graphical representation of the problem to clarify the discrepancy between the correct and incorrect solutions and facilitate the formation of a functional mental model (conceptual knowledge). Thus, the tutor presents two animated trains appearing on opposite sides of the screen, and converging at a point almost in the middle of the screen. They travel at different rates of speed. The problem statement stays up at the top of the screen, and the tutor points out, as it periodically pauses the simulation, what elements should be attended to and when. After Jeremy states that he understands the mapping between the explicated conceptual knowledge, the appropriate equation, and the relevant parts of the word problem, the computer presents an isomorphic word problem. This time he solves it correctly, without any supplemental graphics. The computer allows him to play around with some trains, missiles and boats on his own for a while to test his emerging understanding. He views his "score" of curricular elements acquired, and instruction and learning continue.

The above scenario could describe events in a class-lab 15 to 50 years from now, or just remain a figment of our imaginations. To achieve this future, we need to conduct more systematic research on student modeling, focusing on increased system flexibility and diagnostic accuracy. We also need to adopt some framework and formalism for a more precise specification of models. All this presumes that the student model is the correct focus for developing more intelligent tutoring systems, so we additionally need controlled evaluations testing which modeling techniques are better for which kinds of domains, and even whether or not a student model, in general, is worth the research and development costs.

The ability to diagnose student errors and tailor remediation based on the diagnosis represents the critical difference between intelligent and merely clever computer-assisted instruction. The working definition of computer-tutor intelligence that I'll be using in this paper is that a system must *behave* intelligently, not actually *be* intelligent, like a human. Specifically, an intelligent system must be able to accurately diagnose students' knowledge structures, skills, and/or learning styles using principles, rather than pre-programmed responses, to decide what to do next, and then adapt instruction accordingly (Shute & Psotka, in press).

Generic Intelligent Tutoring System

A student learns from an intelligent tutoring system (ITS) primarily by solving problems--ones that are appropriately selected or tailor-made, and that serve as good learning experiences for that student. The system starts by assessing what the student already knows. This is called the *student model*. The system concurrently must consider what the student needs to learn, embodied in the curriculum (or *domain expert*) as instructional goals. Finally, the system must decide what curriculum element (unit of instruction) ought to be instructed next, and how it shall be presented. This is achieved by the inherent teaching strategy, or *tutor*, in communication with the student model. From all of these considerations, the system selects or generates a problem, then either works out a solution to the problem (via the domain expert), or retrieves a prepared solution. The tutor then compares its solution, in real-time, to the one the student has prepared and performs a diagnosis based on differences between the two. Feedback is offered by the ITS based on issues such as how long it's been since feedback was last provided, whether the student already received some particular advice, what the student's strengths and weaknesses are, and so on. After this, the program updates the student model (the dynamic record of what the student knows and doesn't know). Following these updating activities, the entire cycle is repeated, starting with selecting or generating a new problem.

Not all ITS include these components, and the problem-test-feedback cycle does not adequately characterize all systems. However, this generic depiction does describe many current ITS. Alternative implementations exist, representing philosophical as well as practical differences in their design. For example, the standard approach to building a student model involves representing emerging knowledge and skills of the learner. The computer responds to updated observations with a modified curriculum that is minutely adjusted. This may be called a *microadaptive* approach to modeling, where instruction is very much dependent on individual response histories. An alternative, *macroadaptive* approach involves assessing incoming knowledge and skills, either instead of, or in addition to, emerging knowledge and skills. This enables the curriculum to adapt to both persistent and/or momentary performance information as well as their interaction (see Shute, 1993a, 1993b). I'll now present a student modeling framework designed to aid in the development and comparison of different modeling techniques by providing a standard formalism.

A Framework for Student Modeling

Dillenbourg and Self (1992) outlined a two-dimensional framework and notation for student modeling. Their vertical dimension distinguishes among learner behavior, behavioral knowledge, and conceptual knowledge. This is crossed by the second dimension reflecting the representation of that knowledge--by the learner, the system, and the system's representation of the learner. I modified the framework slightly (see Figure 1) to represent specific knowledge and skill types required during the learning process (i.e., symbolic knowledge, **SK**; procedural skill, **PS**; and conceptual knowledge, **CK**), rather than overt behaviors. The horizontal axis remains basically the same (i.e., learner's representation of the knowledge or skill, **L**; the system's representation of the learner's knowledge, **S/L**; and the system's/expert's representation of the knowledge or skill, **S**). This modified framework reflects the standard microadaptive approach to student modeling.

I also included in the framework a fourth element on the vertical axis reflecting cognitive process measures (or aptitudes). This was added to accommodate individual differences among learners (for macroadaptation), as well as differential cognitive requirements of problems or tasks. Finally, although not shown in the figure, I incorporated a correlated time dimension because, as learning progresses, cognitive demands on the individual vary. For example, working-memory capacity and associative learning skills play an important role early in the learning process in determining the degree of new knowledge and skill acquisition. But over time, these become less important, and other cognitive factors (e.g., perceptual-motor speed) gain importance.

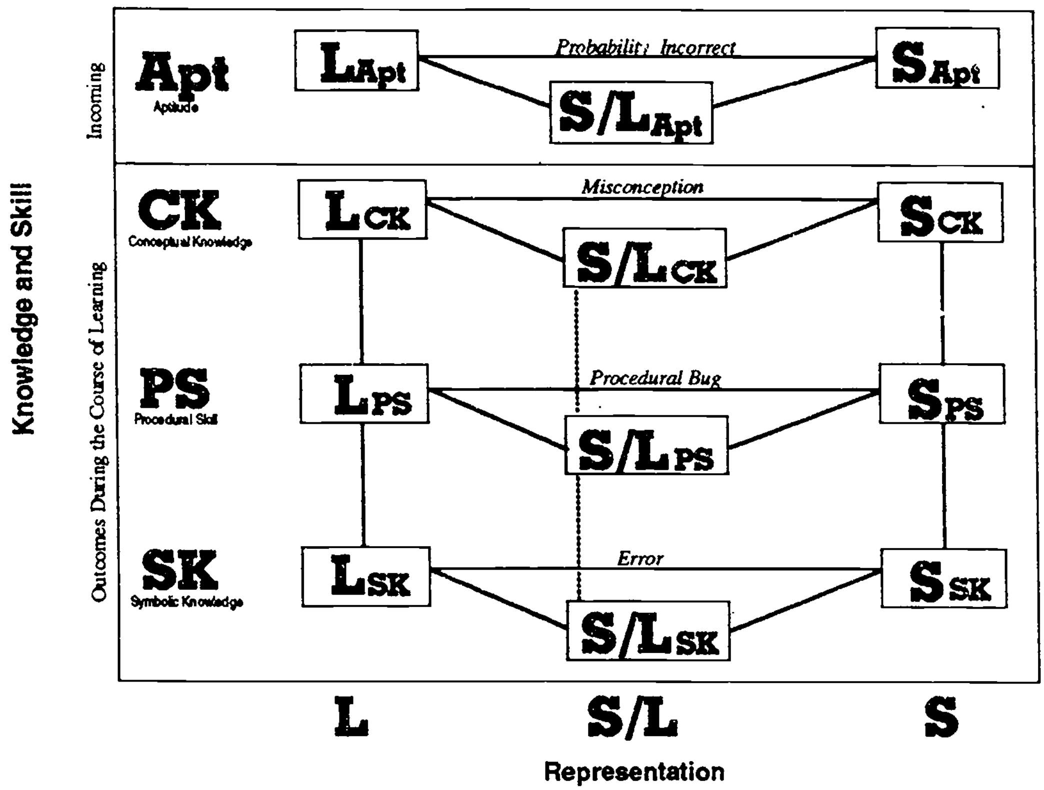


Figure 1: Framework for Student Modeling (adapted from Dillenbourg & Self, 1992)

Let's see how this works. Suppose you were developing a computer program to teach statistical topics, and you were at the part of the curriculum instructing the computation of a measure of central tendency--the MEAN. Your system begins by introducing relevant symbolic knowledge (e.g., Σ, X, N, ΣX/N), then requiring learners to solve problems to demonstrate their acquisition of this new knowledge. Differences between the learner's and the system's representation (or understanding) of the knowledge would show up as *errors* on specific problems, such as failing to recognize the denominator (N) as being a part of the final formula (see bottom section of Figure 1, above). Next, the system directs students to apply their new knowledge by actually computing the MEAN from a set of data (procedural skill). Disparities between the learner's actual solution process (and product) and the system's representation of the correct procedure would be apparent in *procedural bugs*. Finally, learners are taught (and induce), the conceptual knowledge that the MEAN represents the arithmetic average of a set of data, and how that relates to other measures of central tendency (e.g., the location of the MEAN in relation to the MEDIAN and MODE in a skewed distribution). Discrepancies between the learner's and system's conceptual representations (or mental models) can reflect fundamental *misconceptions*.

Another variable believed to influence the probability that a learner will get a specific problem correct or incorrect involves the match between a learner's aptitude (L_{Apt}) and the cognitive requirements of a specific task (S_{Apt}). That is, if the working memory demands/requirements for task X were high (e.g., introduction of a new curricular element requiring the integration of diverse knowledge), and the learner's actual working-memory capacity was low, the predicted probability of success would be low. Cognitive process measures (aptitudes) can easily be assessed. For example, the Cognitive Abilities Measurement (CAM) battery of computerized tests (Kyllonen, Woltz, Christal, Shute, Tirre, and Chaiken, 1990) measures six different aptitudes (i.e., working-memory capacity, inductive reasoning skills, information processing speed, associative learning skills, procedural learning skills, and general knowledge) in each of three domains (quantitative, verbal, and spatial). The battery has been widely-tested and validated, and individual tests can be extracted for different purposes. For instance, if you were teaching a three-dimensional navigational task and wanted to know an individual's spatial aptitude, you could select certain tests for administration (e.g., tests measuring spatial working memory). Differences between the learners actual aptitude (L_{Apt}) and the system's representation of the learner's aptitude (S/L_{Apt}), denotes the construct validity of the aptitude measure (e.g., CAM test validity). Moreover, discrepancies between the system's representation of the learner's aptitude (S/L_{Apt}) and the system's aptitude requirements (S_{Apt}) indicate what kind of remediation is required (e.g., decomposing the current task into more manageable units for a learner diagnosed as having low working-memory capacity).

This general framework also allows for extensions. For instance, the learner's representation of his/her own knowledge (e.g., L/L_{CK}) would denote metacognitive awareness of his or her underlying knowledge structure. I'll now use this framework as a basis for discussing student modeling within a single tutor that attempts to teach a variety of knowledge types to different learners.

A Real-World Attempt to Add Intelligence to a Computer Tutor

Non-Intelligent Version of Stat Lady

Stat Lady (SL) consists of an experiential learning environment and a curriculum that teaches statistical concepts and skills (e.g., DESCRIPTIVE STATISTICS). The curriculum is presented in a relatively fixed order to all learners. The curriculum focused on in this paper was built on a hierarchy of simple to more complex concepts and skills, and consists of three 2-hour modules: (a) data organization and plotting, (b) measures of central tendency, and (c) measures of variability. In addition to student problem solving within the three modules, and the provision of specific feedback, the system also allows learners to engage in elective "extracurricular" activities, such as viewing the on-line Dictionary and Formula Bank, playing around in the Number Factory, or using the Grab-a-Graph features.

The non-intelligent version of SL is often clever, but not intelligent in the classical sense (see working definition, above). The type of pedagogy embedded in the system may be called "near mastery learning." That is, relevant concepts and rules are presented, then SL poses various problems for students to solve in order to demonstrate comprehension of the curricular elements. If a learner continues to get a problem wrong, SL provides three levels of feedback, progressively more explicit. For any particular problem, if the student fails to solve it correctly after three attempts, he or she is given the correct answer. The system thus *presumes* that the learner has actually acquired that concept or skill after explicitly being told. But this could be an erroneous presumption, with problems arising later on when a student tries to learn higher-level skills that have only partially-learned subskills as components. *As the twig is bent, so grows the tree.*

Suppose that two students, Tonya and Nancy, have advanced to the "Measures of Central Tendency" module. Specifically, they are beginning the section concerning the computation of the MEAN. *Stat Lady* begins this topic by identifying the MEAN as the most commonly used measure of Central Tendency, and as being synonymous with the *average*--a term familiar to most students. After presenting these broad conceptual strokes, *Stat Lady* quickly illustrates the concepts with a real-life example (e.g., computing an average grade in Biology class from a group of test scores).

Tonya and Nancy then must obtain some test scores from the Number Factory (an activity that's considered analogous to data collection), and *Stat Lady* walks them through the process of building the formula for the MEAN. Even though they're only in the earliest stages of learning this new concept, *Stat Lady* constantly challenges her students to think for themselves by drawing on bits and pieces of relevant information covered in a previous section of the tutor. For instance, rather than simply giving Tonya and Nancy the formula for the MEAN, she tests their symbolic knowledge (SK) and associative learning abilities by asking them to use a menu of options to create the notation that stands for summing scores in a sample (ΣX). They'd worked with these two symbols before, but never conjointly. Right away, we see some individual differences in learning--Nancy has no problem with the task just described. She enters the correct answer on her first attempt, is congratulated with positive auditory and textual feedback, and moves directly on to actually summing her scores (i.e., a test of procedural skill, PS). Tonya, on the other hand, is struggling. She enters a series of incorrect responses, each of which is followed by the increasingly explicit feedback (and encouragement) mentioned above. *Stat Lady* finally gives Tonya the answer and tells her to go back and enter it, and then Tonya is advanced to the "Computation of the Sum" exercise.

While both students eventually proceed through the identification of "N" as the denominator of the formula, determination of the actual sample size, and the computation of the MEAN of the obtained test scores, an examination of their performance histories reveals two very different stories. Nancy is successful time and again, and even computes the MEAN correctly on her first attempt. *Stat Lady* congratulates her and she moves on to solve colorful problem sets, applying the formula she's just created. In contrast, Tonya has a rough time of it. She never seems to catch on and has to be given the answer to everything except the actual sample size (N), a concept she remembered from the "Data Organization and Plotting" module.

Even a quick illustration such as this makes the inherent problem apparent. Although it seems safe to assume that Nancy has a firm understanding of the MEAN (symbolic knowledge and procedural skill), the same can't be said for poor Tonya. It's *possible* that, at some point during, or after, the presentation of the full formula, all of the puzzle pieces coalesce for Tonya and she actually acquires the desired knowledge and skill, but there's no firm *proof* that this ever happened. Ideally, all students would walk away from a tutor having demonstrated the knowledge level that Nancy evidenced. How can *Stat Lady* be modified to yield more guarantees of successful knowledge and skill acquisition?

Intelligent Version of Stat Lady

The first step in rendering *Stat Lady* intelligent was to perform an extensive cognitive task analysis of the curriculum, employing two subject-matter experts for reliability. This yielded more than 250 curricular elements that were classified into: symbolic knowledge, procedural skills, and conceptual knowledge. Next, these elements were arranged in a hierarchy, relating higher-level knowledge and skills to one another and to successively lower-level knowledge and skills. In the example above concerning the acquisition of symbolic knowledge of the MEAN, correctly specifying the formula presumes familiarity with (if not complete comprehension of) four lower-level SK curricular elements: the summation notation (Σ), variable values (X), total sample size (N), and the way in which these components are combined ($\Sigma X/N$). A more *conceptual* understanding of this concept requires comprehension of frequency distributions, learned earlier, as well as realizing that the MEAN represents one of three measures of central tendency, all related to one another. Moving up the hierarchy, the MEAN also relates to issues of variability, where computing or understanding standard deviation is dependent on successful acquisition of the MEAN (SK, PS, and CK). Unfortunately, the size limitations of this paper make it impossible to depict the entire network.

The intelligent version of *Stat Lady* (ISL) includes a student model. Actually, there are three student models, one each for SK, PS, and CK, but I'll just refer to them collectively as "student model" because they apply the same basic updating heuristics. The student model is represented by a directed graph with values associated with each node/element showing the mastery level of each student. The edges of the graph represent prerequisite knowledge and skills. This is called an "inheritance hierarchy" where different elements are shown along with their "children" or component parts.

Initial values for the model are obtained from students' performance on a comprehensive pretest designed to assess incoming knowledge of *all* curriculum elements resulting from the cognitive task analysis. Based on the assessment of pretest performance, subjects are then placed into the appropriate part of the curriculum (macroadaptation). For instance, if a learner correctly answered all items relating to frequency distributions, proportions, percentages, MEAN, MEDIAN, and MODE, those items would all receive a "1" in the preliminary student model. If he or she failed all items relating to variance, standard deviation, skewness, and kurtosis, those values would retain their initialized values of "0" and that individual would start the tutor at the section of the curriculum containing the lowest level of curricular element containing a value of "0."

Fuzzy student modeling variables may be associated with each curricular element. Values assigned to these variables range from 0 to 1 representing knowledge states, from "most likely unknown" to "most likely known" (see Derry & Hawkes, 1993; Lesgold, Eggan, Katz, & Rao, 1992). The preliminary student model, instantiated with binary (0/1) data from the pretest performance, is then continuously updated during the course of learning--at the conclusion of each problem solving endeavor. Updating (i.e., upgrading and downgrading values) occurs first at the lowest levels (e.g., individual components), then is propagated up the hierarchy. A student is moved on to the next appropriate unit of instruction upon exceeding a student-model value of .89, the mastery criterion for each curricular element.

Following placement in the curriculum from pretest assessment, *Stat Lady* introduces the current topic, and students proceed to solve related problems. The student can correctly solve the problem on his/her first try, or incorrectly solve it. As described above, SL has three levels of feedback for incorrect responses, thus a person can receive 0, 1, 2, or 3 hints for a given problem. The probability of successful acquisition of that problem is simply defined as: [1 - (misses/3)] * Pretest constant (see Figure 2). The Pretest constant is: if correct, multiply by .91; if incorrect, multiply by .89. Learners therefore won't be advanced if they are not quite successful (< .9). These two differential weights (.91, .89) are used because, if a learner correctly solves an item on the pretest and is also correct on her first attempt during problem solution, the evidence suggests that she knows the item and can be advanced given the above-threshold value ([1 - 0/3)] * .91 = .91). This is slightly higher than the case where a person misses a particular item on the pretest, but solves the problem correctly without any hints ([1 - 0/3)] * .89 = .89) during the tutor. In the latter case, the learner does not advance to the next unit of instruction because it's easier to solve a problem following some instruction compared to initially coming up with an answer on one's own, as required in the pretest. Beginning values for the eight different conditions are shown below.

Problem Performance
Number of Misses (Number of Hints)

Pretest Performance	0	1	2	3
1	**.91**	**.61**	**.30**	**0**
0	**.89**	**.59**	**.30**	**0**

Figure 2: Student Model of One Curricular Element Following a Problem-Solving Episode

For values less than .9, the system presents a fresh problem, and, depending on the learner's performance, the new value is combined with the old to yield a composite success score. To illustrate, suppose Nancy failed all pretest items relating to curriculum element X. When placed in the curriculum to learn about X, she required one hint before successfully solving the problem. Her initial student model value would be: .59 (i.e., (1 - 1/3) * .89). Another problem is presented, and this time she gets it correct without any assistance, so her new score is 1 (1 - 0/3 = 1) (Note: the pretest function only enters into the equation with the first problem solving episode). The combined success score at this point updates to: (1 + .59)/2 = .8. While close, this value is still slightly below criterion performance. Thus a third problem is presented, and Nancy completes the problem effortlessly (with no hints). The combined score now reaches threshold (.8 + 1)/2 = .9, and Nancy begins the next topic. In contrast, a learner beginning with a value of 0 would require a minimum of 4 fresh problems before advancement, providing he or she was successful on each of those (i.e., start at 0, then update to .5, .75, .88, .94). Slower students may take even more problems, carefully tailored to fit the specifically-diagnosed deficits or misconceptions.

These values are also propagated up the inheritance hierarchy to provide preliminary estimates for complex curricular elements that contain element X as a component knowledge or skill. As mentioned, the STANDARD DEVIATION requires an understanding of the MEAN in addition to an understanding about distributions, deviations from the MEAN, sum of squares, and so forth. The current student-model value for MEAN, then, becomes part of the preliminary value for the standard deviation, along with other information, such as pretest performance data, complexity of the curricular element, and additional information concerning the learner's aptitude (the last two have not yet been implemented). A curricular element's difficulty or complexity may be based on indicators such as the number of required elements and operators, expert ratings of difficulty, and empirically-determined indices of element difficulty. The learner's aptitude may be assessed via the CAM battery, mentioned earlier.

The student model is visible to the student in the form of an on-line Report Card, presented as a bar graph showing the level of mastery the system believes the student has achieved (i.e., S/L_{SK}, S/L_{PS}, S/L_{CK}). The Report Card represents concepts and skills at a global level (e.g., SK of the MEAN, PS in computing the MEAN), then is further decomposable into individual elements (e.g., success on Σ, N, X, $\Sigma X/N$). Additional records are maintained on students' usage of the elective tools, referred to earlier (e.g., number of times they accessed the on-line Dictionary). This information is inspectable as well, and eventually will work its way into the formal student model. That is, a person who supplements his/her tutorial instruction by engaging in self-initiated reading of the hypertext Dictionary, for instance, is probably more successful in knowledge and skill acquisition (compared to passive learners) given their active, motivated learning behaviors.

Discussion

There is a large cost associated with incorporating a student model into a tutoring system. This raises two important questions: (1) How much, and what kind of, information about a learner is required to adequately diagnose knowledge and skill acquisition and subsequently tailor instruction to the learner's needs? (2) What is the payoff of increasing a system's adaptability? Sleeman (1987) has argued that "... if one takes seriously the findings of the ATI work of Cronbach and Snow (1977), it would appear that there is little likelihood of producing instruction that is uniquely individualized" (p. 242). The key word in this statement is "uniquely." An exhaustive characterization of a learner would probably not warrant the effort and expense in terms of increases in final outcome. But the empirical question remains: How much is enough? In other words, does inclusion of a student model in a tutor enhance overall learning outcome or efficiency? There is equivocal evidence in the literature concerning these issues. In some cases, researchers have reported no advantage of error remediation in relation to learning outcome (e.g., Bunderson & Olsen, 1983; Sleeman, Kelly, Martinak, Ward & Moore, 1989), whereas in others, some advantage has been reported for more personalized remediation (e.g., Anderson, 1993; Shute, 1993a, 1993b; Swan, 1983).

Another question asks whether the student model is even the right framework around which to build good learning systems. Derry and Lajoie (1993) presented several reasons why the student modeling paradigm is problematic. Among the more compelling reasons cited were that: (a) In complex domains, the student model can not specify all possible solution paths, nor determine all possible "buggy" behaviors, (b) Reflection and diagnosis should be performed by the student, not the tutor, and (c) Model-tracing is only applicable to procedural learning, but the focus should be on critical thinking and problem solving. The approach to student modeling taken in this paper does not attempt to delineate all possible solution paths or buggy behaviors. Rather, the information about curriculum elements, derived from the cognitive task analysis and arranged in an inheritance hierarchy, provides a basis for inferences about what knowledge and skills have been acquired and to what degree. Furthermore, this approach models not only procedural skill acquisition, but also symbolic and conceptual knowledge acquisition.

Another contribution of this approach is the inclusion of aptitude in the equation as a predictor of subsequent learning. That is, ATI research provides information about initial states of learners that can be applied in *macroadaptive* instruction (e.g., selection of Jeremy's graphics emphasis in the opening scenario, or location in the curriculum that a learner should be placed to maximize instructional efficiency). Subsequently, *microadaptive* instruction can be used as a response to particular actions (e.g., selection of the next small unit of instruction to be presented based on a specific response history). Initial states may be characterized by an aptitude profile, then microadaptive instructional systems can focus on strengths, circumvent weaknesses, or highlight deficits to be remediated.

The two versions of *Stat Lady*, described herein, provide the basis for an experiment testing the degree to which inclusion of a student model may (a) enhance learning outcome measures, and/or (b) improve learning efficiency. Approximately 400 subjects will be run in an upcoming experiment (August, 1994) testing the efficacy of this student modeling approach. Subjects will be randomly assigned to one of the two versions of *Stat Lady* (intelligent vs. non-intelligent). An on-line pretest will be administered, students will proceed through their respective tutors, and then a posttest will be given. We will be looking at pretest to posttest changes in scores, learning rates (the systems are both self-paced), as well as any aptitude by treatment interactions. Finally, this study will provide the basis for a cost-benefit analysis and some preliminary answers to burning questions about the worth of student modeling.

References

Anderson, J. R. (1993). Rules of the mind. Hillsdale, NJ: Lawrence Erlbaum Associates.
Bunderson, C. V., & Olsen, J. B. (1983). Mental errors in arithmetic skills: Their diagnosis in precollege students (Final project report, NSF SED 80-125000). Provo, UT: WICAT Education Institution.
Cronbach, L. J., & Snow, R. E. (1977). Aptitudes and instructional methods: A handbook for research on interactions. New York: Irvington.

Derry, S. J., & Lajoie, S. P. (1993). A middle camp for (un)intelligent instructional computing: An introduction. In S. P. Lajoie & S. J. Derry (Eds.), Computers as cognitive tools (pp. 1-11). Hillsdale, NJ: Lawrence Erlbaum Associates.

Dillenbourg, P., & Self, J. (1992). A framework for learner modelling. Interactive Learning Environments, 2(2), 111-137.

Derry, S. J., & Hawkes, L. W. (1993). Local cognitive modeling of problem-solving behavior: An application of fuzzy theory. In S. P. Lajoie & S. J. Derry (Eds.), Computers as cognitive tools (pp. 107-140). Hillsdale, NJ: Lawrence Erlbaum Associates.

Kyllonen, P. C., Woltz, D. J., Christal, R. E., Shute, V. J., Tirre, W. C., & Chaiken, S. (1990). CAM-4: Computerized battery of cognitive ability tests. Unpublished computer program, Brooks Air Force Base, TX.

Lesgold, A., Eggan, G., Katz, S., & Rao, G. (1992). Possibilities for assessment using computer-based apprenticeship environments. In J. W. Regian & V. J. Shute (Eds.), Cognitive approaches to automated instruction (pp. 49-80). Hillsdale, NJ: Lawrence Erlbaum Associates.

Self, J. A. (1993). Model-based cognitive diagnosis. User-Modeling and User-Adapted Interaction, 3, 89-106.

Shute, V. J. (1993a). A macroadaptive approach to tutoring. Journal of Artificial Intelligence and Education, 4(1), 61-93.

Shute, V. J. (1993b). A comparison of learning environments: All that glitters... In S. P. Lajoie & S. J. Derry (Eds.), Computers as Cognitive Tools (pp. 47-74). Hillsdale, NJ: Lawrence Erlbaum Associates.

Shute, V. J., & Psotka, J. (in press). Intelligent tutoring systems: Past, present, and future. To appear in D. Jonassen (Ed.), Handbook of Research on Educational Communications and Technology. Scholastic Publications.

Sleeman, D. H. (1987). PIXIE: a shell for developing intelligent tutoring systems. In R. Lawler & M. Yazdani (Eds.), AI and education: Learning environments and intelligent tutoring systems (Vol. 1, pp. 239-265). Norwood, NJ: Ablex Publishing.

Sleeman, D. H., Kelly, A. E., Martinak, R., Ward, R. D., & Moore, J. L. (1989). Studies of diagnosis and remediation with high school algebra students. Cognitive Science, 13(4), 551-568.

Snow, R. E. (1990). Toward assessment of cognitive and conative structures in learning. Educational Researcher, 18(9), 8-14.

Swan, M. B. (1983). Teaching decimal place value: A comparative study of conflict and positively-only approaches (Research Rep. No. 31). Nottingham, England: University of Nottingham, Sheel Center for Mathematical Education.

Acknowledgements

I would like to sincerely thank several people who have been working with me on this project: Roger Dubbs, Kevin Gluck, Lee Gugerty, Pat Kyllonen, Nancy Lefort, Dave Logan, and Wes Regian. The research reported in this paper was conducted by personnel of the Armstrong Laboratory, Human Resources Directorate, Brooks Air Force Base, Texas. The opinions expressed in this article are those of the author and do not necessarily reflect those of the Air Force.

Correspondence concerning this paper should be addressed to Valerie J. Shute, AL/HRMI, 7909 Lindbergh Drive, Brooks AFB, Texas 78235-5352.

From Augmentation Media to Meme Media

Yuzuru Tanaka
Electrical Engineering Department, Hokkaido University, Sapporo, 060, JAPAN
tanaka@huee.hokudai.ac.jp

Abstract: Computers as meta media are now evolving from augmentation media vehicles to meme media vehicles. While an augmentation media system provides a seamlessly integrated environment of various tools and documents, meme media system provides further functions to edit and distribute tools and documents. Documents and tools on meme media can easily replicate themselves, recombine themselves, and are naturally selected by their environment, namely the society of their authors and users. Their accumulation in their users' community will form a meme pool, which will bring rapid evolution of documents and tools. The IntelligentPad architecture provides a standard framework called a pad that works as a meme medium. When applied to microworlds, it will bring more flexibility to the environments, and more opportunities of creative thinking to their users.

Media for Creations and Evolutions

Today's personal computers are considered as augmentation media that provide us with various tools for entertainment and thought. They provide tools to augment our capabilities. However, augmentation alone can not stimulate our creativity. Creativity requires new media different from augmentation media. What kind of media can stimulate creativity? This question has been motivating the IntelligentPad research project in Hokkaido University since 1987. Creation yields evolution, and no other than evolution can activate further new creation. Biological evolution is based on genes. We require similar genetic media both for the cultural and technological evolution of creative works on computers, and for the educational evolution of their users' creativity. Such media should be able to replicate themselves, to recombine themselves, and to be naturally selected by their environment. They may be called meme media since they carry what R. Dawkins called "memes" (Dawkins, 1976). Their environment here means the society of their producers and consumers, namely, authors and users. M. Stefik pointed out in 1986 the importance of understanding and building an interactive knowledge medium that embodies the characteristics of memes (Stefik, 1987).

Children playing with the toy blocks Lego easily create various types of vehicles, towns, and amusement parks. As G. Tarde, a French sociologist in the 19th century, thought, creations start from imitations in a broad sense (Clark, 1969). Lego has been keeping to provide new customers with various new construction kits for vehicles, towns and amusement parks so that they can learn construction patterns through their playing experiences with these kits. After the imitating stage, they gradually become able to recombine partial imitations of different originals, or further to add their original patterns to these recombinations. The standard connection interface of Lego blocks allows arbitrary combinations of various blocks from different construction kits. Thus, the world of Lego blocks rapidly expands and evolves themselves. Creation and evolution require such meme media that make both the decomposition of existing objects and the composition of new ones as easy, as direct, and as instantaneous tasks as the editing of documents.

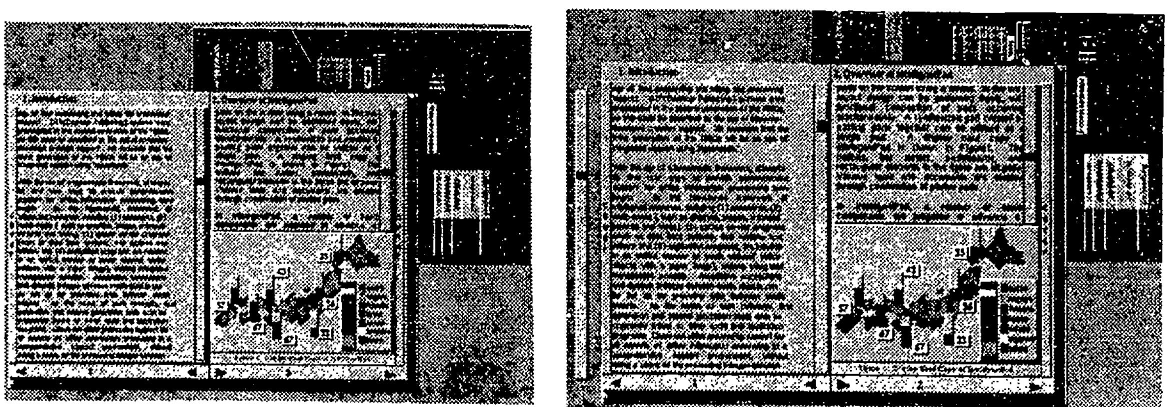
While personal computers have dramatically simplified the editing of multimedia documents, they can not yet allow us to easily edit existing tools to create new tools. The IntelligentPad system architecture that has been developed at Hokkaido University for these 7 years uniformly represents both documents and tools as pads, i.e., reactive media objects that look like paper sheets on the display screen (Tanaka and Imataki, 1989; Tanaka, 1989, 1991; Tanaka et al., 1992). Pads work as meme media for the editing, the replication, and the natural selection of various types of documents and tools.

IntelligentPad as Meme Media

A pad in the IntelligentPad consists of a display object and a model object. Its display object defines both its view on the display screen and its reaction to user events, while its model object defines its internal state and behavior. Wide varieties of intellectual resources can be represented as pads. Among them are multimedia documents, system utilities, application programs, and user environments. Among system utilities are E-mail utilities, file utilities, and database utilities.

In an IntelligentPad system, you can easily compose any document or any tool by directly pasting some pads on top of another. Such a paste operation simultaneously defines both the layout of its components in the composed pad and the functional linkage among component pads. Composite pads are also simply referred to as pads. When the distinction is necessary, pads that are not composite pads are referred to as primitive pads. You may use paste operations in arbitrary ways, for example, to overlay multiple translucent pads of the same size, or to arrange multiple pads on the same base pad. Users can easily replicate any pads, paste pads on another, and peel a pad off a composite pad. These operations can be equally applied to both any primitive pads and any composite pads.

Figure 1 (a) shows a bookshelf and books that are all constructed by pasting various primitive pads. The open book defined as a composite pad shows a text with a scroll bar on the left page and a



(a) Composite pads define a bookshelf and books.

(b) The scroll bar in the text page is replaced with a bar meter in the map.

Figure 1 The editing of an existing composite pad.

map of Japan with several bar meters on the right page. This scroll bar pad and these bar meter pads actually share the same function that detects the mouse location and sends the underlying pad a value between 0 and 1 depending on the detected relative location. Therefore, you may replace the scroll bar with a copy of these bar meters (Figure 1 (b)). When you paste a bar meter on the text pad, you have to connect this bar meter to the text-scrolling function of the underlying text pad. You can specify this connection just by selecting the scroll slot from the list of slots defined by the text pad.

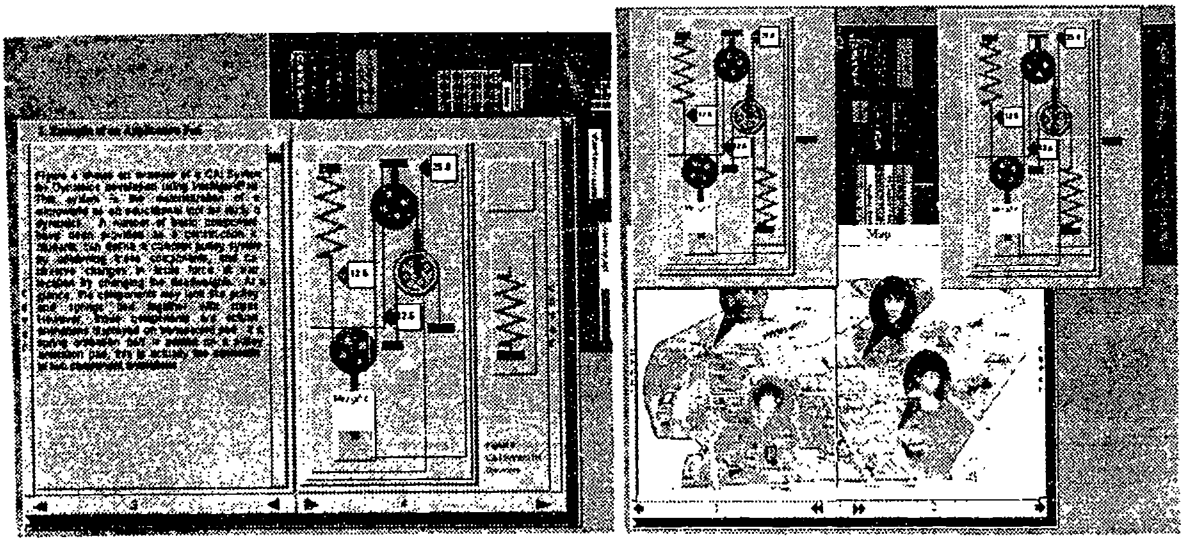
The application-linkage interface of each pad is defined as a list of slots. Each slot can be accessed either by a **set** message **set** <*slot_name*> *value* or by a **gimme** message **gimme** <*slot_name*>. Each of these two messages invokes the respective procedure attached to the slot. Each slot s_i may have two attached procedures, $proc_{i,set}$ and $proc_{i,gimme}$. The default for $proc_{i,set}$ stores the parameter value into the slot register, while the default for $proc_{i,gimme}$ returns the slot register value. Its slots and attached procedures define the internal mechanism of each pad. They are defined by its developer.

When a pad P_2 is pasted on another pad P_1, the IntelligentPad constructs a linkage between them. This defines a dependency from P_1 to P_2. The pad P_2 becomes a subpad of P_1, while P_1

becomes the master pad of P_2. No pad may have more than one master pad. If P_1 has more than one slot, we have to select one of them to associate it with P_2. This selection can be specified on a connection sheet. The selected slot name is stored in a standard slot of the subpad P_2 named *connectslot*. A subpad can send either **set** $\uparrow$ *connectslot* **<value>** or **gimme** $\uparrow$ *connectslot* message to its master, while the master pad, when its state is changed, can send some of its subpads an **update** message without any parameter to propagate an update event. The up arrow before *connectslot* means that the slot name stored in *connectslot* becomes the real parameter. IntelligentPad allows us to disable some of these messages, which can be specified on the connection sheet. The two messages **set** s v and **gimme** s sent to a pad P are forwarded to its master pad if P does not have the slot s.

Besides the three standard messages, any pad can send some standard messages for geometrical operations to its master as well as to its subpads. Among them are move, copy, delete, hide, show, open, close, resize, and paste messages. The set of these standard messages defines a standard interface between pads.

Figure 2 (a) shows other pages of the same book. Pulleys and springs on the right page are



(a) Springs are pulleys that are connected to each other are animated by transparent pads.

(b) A pulley-and-spring pad is put on a FieldPad, and taken its shared copy. Picture pads on a campus map can send any pads to a priori specified workstations.

Figure 2 A microworld of springs are pulleys and the event-sharing by the FieldPad.

animated by transparent pads. By pasting these pads together, you can easily connect animated springs and pulleys. Two pads pasted together automatically readjust the location of their animated parts so that these parts may work as if they are directly connected. These springs and pulleys constitute a construction set for a microworld (Papert, 1980), where a learner can not only play with a given set of objects, but also decompose these objects and recombine their components to create new objects.

Any pads accept copy and shared-copy requests. Shared copies of the same pad share the same state, i.e., the same model object. The state of a composite pad is defined as the state of the base pad. Shared copies, however, can not share a user event applied to one of them unless it only changes the shared state. A user event may change the pad view without changing its state. For example, it may change only the location of a component pad in a composite pad. Hence, we require an event sharing mechanism as an independent primitive function. In the IntelligentPad architecture, every independent function should be implemented as an independent pad so that its generic function may be applicable to any pads. The pad on the right of the pulley-and-spring pad in Figure 2 (a) works as such an event-sharing pad. We call it a FieldPad since it represents the field of user events. In Figure 2 (b), the pulley-and-spring pad is pasted on the enlarged FieldPad, and the whole pad is duplicated. The two copies share every user event applied to it.

In Figure 2(b), a campus map of our university is pulled out of the bookshelf. It has pictures of several persons at different sites. Each picture is actually a pad clipped into that shape. This clipping facility is also provided as a pad called a MaskPad. When a pad is pasted on a MaskPad, it is clipped into the shape of this MaskPad. Each picture pad on the campus map can transport any pad to an a priori specified IP address. If you want to send one of the shared copies in Figure 2 (b) to one of the persons in the campus map, you can just drag this copy to the picture of this person and drop it there. The copy is automatically sent to the workstation of that person. Shared copies distributed over a network can still keep sharing the same state.

From Object-Oriented Architectures to Media-Based Architectures

Current computers treat various types of intellectual resources. Among them are multimedia documents, system utilities, application systems, and user environments. Unless they can provide a dedicated functional linkage between any pair of these different types of resources, they fail to integrate these resources (Figure 3 (a)). By a functional linkage is meant not only a data communication linkage but also an application linkage through message exchanges. Different pairs require different types of linkages. If there are n different types, we require $O(n^2)$ different types of linkages. This is the essential difficulties of open integration systems that are open to the future addition of new intellectual resource types.

The IntelligentPad architecture solved this problem by separating media from their contents, and standardizing the logical structure and the interface of the media. Media of a certain type in general play their most important role in providing a uniform access protocol for various types of intellectual resources. Books are the most typical example. They have a long history for the development of their common structure starting with a front cover, followed by contents, then a body with hundreds of pages, indices, and then ending with a rear cover. Books are organized in this way to provide their readers with a uniform access protocol. While media of the same kind share the same organization structure, they can contain different structures of information as their contents. The same idea was adopted by the IntelligentPad architecture. Each primitive pad consists of its shell and its content. Its shell defines its standard media structure and interface. It is up to the developer of each pad how the contents is implemented in the standard shell. While the IntelligentPad is based on an object-oriented architecture, it further restricts its architecture. Therefore, we call such architectures media-based architectures to distinguish them from conventional object-oriented architectures (Figure 3 (b)).

In a media-based architecture, only one type of functional linkages is used to connect any pair of shells (Figure 3 (b)). In the IntelligentPad, each shell has an arbitrary number of jacks called slots. It also has a single pin-plug to connect itself to one of the slots of another shell. The shell architecture and the standard linkage facility are provided by the kernel of the IntelligentPad systems. Neither users nor pad developers have to worry about them.

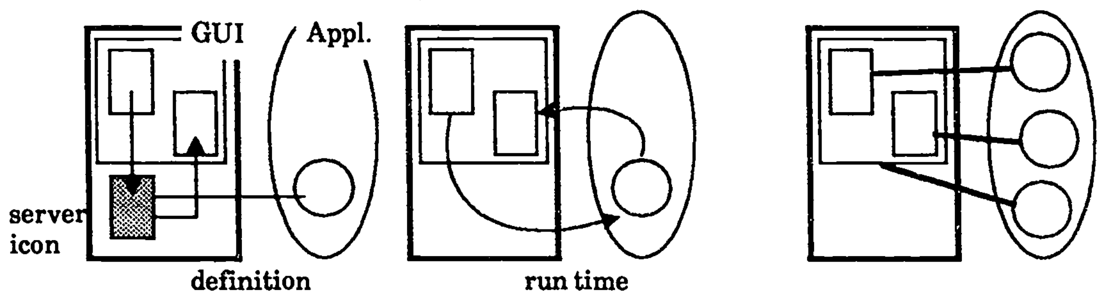
(a) object-oriented architectures (b) media-based architectures

Figure 3 Object-Oriented Architectures and Media-Based Architectures

Pad Synthesis as the Application Linkage through the Embedding

Interapplication communication mechanisms provided by various graphical environments can be classified into the following 4 categories; (1) Cut-and-Paste, (2) Drag-and-Drop, (3) Object Linking, and (4) Embedding. The first two may require no explanations. Object Linking adopts graphical links to interrelate two different application objects. Embedding adopts a document-oriented model. In a document-oriented model, a compound document created by a client application may contain various component objects linked to different server applications.

The embedding mechanisms are further classified into two categories; (1) Widget Embedding, and (2) Media Embedding (Figure 4). In the widget embedding, the link between a component and



(a) Widget Embedding (b) Media Embedding

Figure 4 The widget embedding and the media embedding.

a server application is not fixed, and hence it needs to be defined by the object linking mechanism. Besides, a single server application can be differently linked to more than one components. The NeXT's InterfaceBuilder falls into this category. In media embedding, the link between a component and a server application is fixed and never changes wherever this component is embedded. This fixed pair defines a media object. Besides, in media embedding, the embedding of a media object into another media object defines an application linkage between these two media objects. The IntelligentPad architecture adopts the media embedding mechanism. Suppose you make a copy of the object with some embedded objects. In the widget embedding, this copy can not inherit its original's relations to the server applications and hence loses the original function. In the media embedding, however, this copy inherits its original function.

Only the media-embedding mechanism enables us to edit tools to define new functions as if we edit compound documents.

A Pad Pool as a Meme Pool and Its Search

The IntelligentPad architecture aims at the exchange and the distribution of pads across different platforms, which enables its users society to form an open market of pads. Some pads will be priced, and others will be free. Each user can easily import any pad from an open market into his own environment through networks or via off-line storage media. He may decompose this imported pad, paste its arbitrary subcomponent on top of another owned in his environment, and create a new pad. This new pad can be again published into the market.

No more than 10,000 programmers, each capable of developing only three primitive pads in a year, can add 30,000 new primitive pads every year to a pile of pads in the society. If 50,000,000 people access the pad pool in the society, and if each of them edits pads to create 5 new composite pads every year, we can expect the increase of 250,000,000 pads every year in the pad pool. These pads are genes for the computers and the networks in our society. Pads can be recombined like genes. An addition of a new pad into a composite pad works as mutation. The accumulation of pads in a market forms a gene pool. Useful pads are more widely distributed by the users replicating them for their own use. This is a natural selection mechanism. Pads work as memes. The pad pool shared by our society works as a meme pool, and promises rapid evolution of various types of intellectual resources represented by pads.

Such a society will not face the so-called software crisis. Instead, it will face new kinds of problems. "How can we rapidly distribute among users the intellectual resources they want?' "How can users search the expanding shared pool of intellectual resources for what they need? " These are exactly the same questions we have now in our consumer culture flooded with various consumer products. Users and intellectual resources respectively correspond to consumers and consumer goods. Meme media will bring us consumer societies and consumer cultures of intellectual resources, which requires new services for the distribution, management and retrieval of intellectual resources.

The IntelligentPad architecture provides four kinds of access methods: (1) a visual catalog of pads for browsing, (2) hypermedia networks for navigation, and both (3) form bases and (4) pad bases for quantified retrieval of pads. Any pads can be managed by these four search methods.

This means that we can even store and retrieve CSCW environments constructed with a FieldPad, CAI environments, or computer games, both constructed with a StagePad, as well as documents, charts, tables, and ordinary desk work tools. A stagePad is used to simulate a sequence of user tasks using multiple pads. Form bases cannot meet our needs to manage all types of pads. IntelligentPad provides the PadBasePad to meet such needs. A pad base is a database in which stored records are all pads. In our daily conversation, we specify what we want by its name if we know the name. Otherwise, we partially specify it either by its category, by its substructure, by its context, or by some combination of these. We may describe fish as things in water. This specifies its context. We may describe a calculator as a thing with more than 10 buttons and a digital display. This specifies its substructure. Pad bases allow us to specify the pads we want by any of these different methods.

Remarks on CAI Applications of IntelligentPad

The IntelligentPad architecture works as a meta tool. It provides tool developers with a standard pad framework to program each tool component as a pad, a large library of pads, and an open environment to export and import various pads to and from other IntelligentPad environments. When applied to microworlds (Papert, 1980), it can provide not only tools and objects that their users can easily combine, but also their construction kits, which enable the users to customize or to decompose the given tools and objects, or further to invent new tools and objects. Besides, a user can easily expand his microworld by importing new tools and objects from any other different microworlds.

The IntelligentPad architecture also provides a powerful mechanism called a proxy pad. A proxy pad works as an interface to an external object such as a simulation program running on either the same machine or the different machine connected by a network, a database system, a computer-controlled device like a VCR, a computer controlled measurement tool, or an industrial plant. Proxy pads enable the microworlds to assimilate external objects, especially objects in the users' real world.

The IntelligentPad will surely work as a meta tool for the development of what Ferguson called "exploratory environments" (Ferguson, 1992).

The IntelligentPad architecture is now implemented in several languages. The Smalltalk 80 version provides more than 400 primitive pads. The SmalltalkAgents version allows concurrent operations of pads. It has made the Macintosh windows and the Toolbox resources work as pads. These two versions are already available. A new version developed in Interviews and C++ will also become available this June.

References

Clark, T. N. (1969). *Gabriel Tarde on Communication & Social Influence*, University of Chicago Press.

Dawkins, R. (1976). *The Selfish Gene*, Oxford Univ. Press.

Ferguson, D. L. (1992). Computers in Teaching and Learning: An Interpretation of Current Practices and Suggestions for Future Directions, In *New Directions in Educational Technology*, (ed. by E. Scanlon and T. O'Shea), Nato ASI Series F-96, Springer-Verlag, 33-50.

McLuhan, M. (1964). *Understanding Media, The Extensions of Man*, McGraw-Hill, New York.

Papert, S. (1980). *Mindstorms : Children, Computers and Powerful Ideas*, Basic Books.

Stefik, M. (1986). The Next Knowledge Medium, *The AI Magazine*, 7 (1), 34-46.

Tanaka, Y. (1989). A Toolkit System for the Synthesis and the Management of Active Media Objects,In *Proc. of the 1st Int'l Conf. on Deductive and Object-Oriented Databases*, Kyoto, Dec., 76-94.

Tanaka, Y. (1991). A Synthetic Dynamic-Media System, In *Proc. of Int'l Conf. on Multimedia Information Systems*, Singapore, 299-310.

Tanaka, Y., and Imataki, T. (1989). IntelligentPad: A Hypermedia System allowing Functional Composition of Active Media Objects through Direct Manipulations, *IFIP '89*, San Francisco, Aug., 541-546.

Tanaka, Y., Nagasaki, A., Akaishi,M., and Noguchi, T. (1992). A Synthetic Media Architecture for an Object-Oriented Open Platform, *Personal Computers and Intelligent Systems, IFIP Congress 92*, Vol III, (ed. F. H. Vogt), North Holland, 104-110.

FULL PAPERS

Design Issues in a Collaborative Intelligent Learning Environment for Japanese Language Patterns

Gerardo AYALA and Yoneo YANO
Dept. Information Science and Intelligent Systems
University of Tokushima , 2 - 1 Minamijosanjima-Cho, Tokushima 770, Japan
E-Mail: ayalasan@nl5.is.tokushima-u.ac.jp

Abstract: This paper presents the main issues concerning the design and development of GRACILE, a collaborative intelligent learning environment in the domain of Japanese language patterns and expressions. We present an environment designed to allow learning goals communication, learning activities negotiation, and collaboration between learners. From a group of student models, a potential development level is represented for each learner together with her/his group-based knowledge frontier, which is defined with respect to the knowledge of other learners and the more complex language patterns containing basic patterns which the system believes have already been internalized by the student.

The environment provides the learners with a toolbox for dialogue construction, including dictionaries. A set of dialogue agents, whose capabilities are the construction and appropriate use of language patterns and expressions in different dialogue situations, may also be requested for help. During sentence construction, learners can be assisted by other learners.

Introduction

Japanese language students in Japan face communication problems concerning the appropriate use of language in specific situations, as well as in the comprehension and production of written Japanese. This problem is especially relevant due to the increasing number of people studying Japanese. Nevertheless, research projects on the development of Japanese language learning software for foreigners are few (Hayashi & Yano, 1992; Kusanagi, 1992). CALL systems concerning both grammatical and social aspects of language use are still uncommon (Chanier & Renié, 1993).

Current research on intelligent collaborative systems proposes learning environments that facilitate learning through collaboration. The design of learning environments where students can learn more efficiently and effectively, sharing their knowledge, authority and responsibility in the learning process has been proposed (Chung, 1991). Distant Open Learning systems are approaching the development of systems for the social construction of shared knowledge (Clement, Vieville & Vilers, 1992). Recently, the concept of ICLS (Intelligent Collaborative Learning Systems) (McManus, 1993) appeared, referring to the approach based on the conjunction of ITS and CSCW (Computer Supported Collaborative Work) technologies.

Collaborative Learning by Dialogue Construction

It is considered that the interaction with a second language in its written form represents input from which the learners can discover grammatical regularities more easily than in its spoken form (Krashen & Terrel, 1983; Carrol, 1986). Grammar and writing are important to help the learner rise to a higher level of speech development (Vygotsky, 1986). In second language learning research, there is evidence that techniques which lead to *creative language use* are more appropriate (Carrol, 1986), and it is also accepted that learning environments should provide situations where the learners understand the usage of knowledge through activity. Since the learner who is involved in the authentic activity of writing needs a real audience, this should be provided by the other learners in a collaborative environment (Pacey, 1990). We are of the opinion that by constructing a dialogue together with GRACILE, the learners are able to comprehend and construct sentences, thereby discovering language grammar regularities and understanding the relation between Japanese language patterns/ expressions and the situations where they are appropriately used.

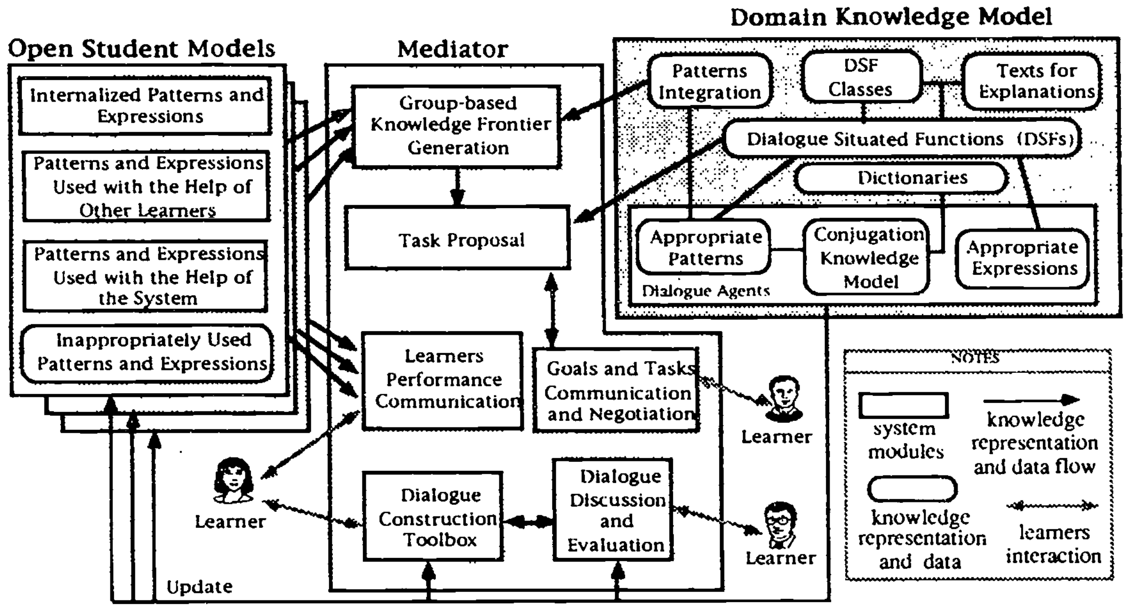


Figure 1. Configuration of GRACILE

GRACILE

We believe the learning of Japanese language patterns and expressions can be more effective as a social activity, where learners play the roles of writers and readers; constructing sentences for a common dialogue, helping each other, and negotiating the domain knowledge to be practiced and its appropriate use. GRACILE's main component is the Mediator, an intelligent agent that cooperates with each learner, generating the respective student's *group-based knowledge frontier*, which represents the knowledge that can be integrated from knowledge acquired by the learner and which has been used by other members of the learning group. The concept of group-based knowledge frontier thus represents the students learning and collaboration opportunities. In this way, each Mediator moderates the negotiation and coordination of the student's learning activities. The architecture of GRACILE is presented in figure 1.

Modeling the Domain Knowledge

Dialogue Situated Functions

A dialogue, as a sequence of sentences, can also be considered as a sequence of accomplished communication goals. Each language pattern and expression has a communicative function that corresponds to a particular communication goal. We call these functions *dialogue situated functions* (hereafter referred to as DSFs), since they appear in a situation of a dialogue. Communication goals are then expressed in terms of DSFs such as "affirmative request", "ask for attention", "show agreement", "apologize", etc. Figure 2 presents an example of a dialogue constructed as a sequence of DSFs expressed by sentences constructed from language patterns and expressions. The domain knowledge is organized by relating the knowledge representation of language patterns and expressions to the corresponding DSFs where appropriately used. Based on the results of a questionnaire to foreigners studying Japanese, we have determined a set of DSFs considered to be representative of the communication goals for a Japanese language student. These DSFs are organized into classes which facilitates their management by the system and the learner.

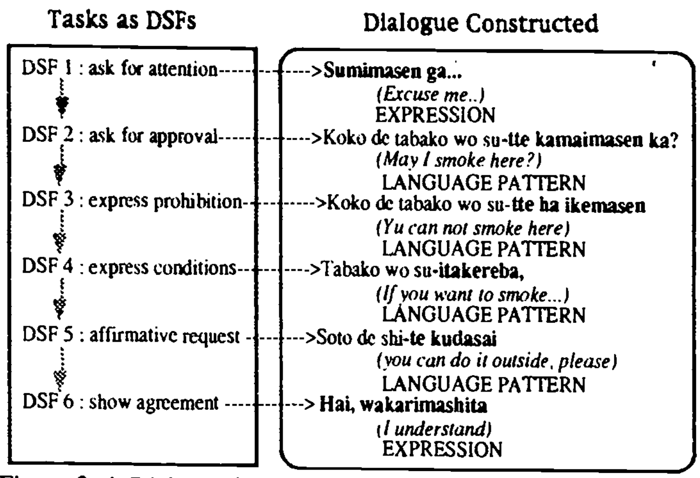


Figure 2. A Dialogue in Terms of Dialogue Situated Functions

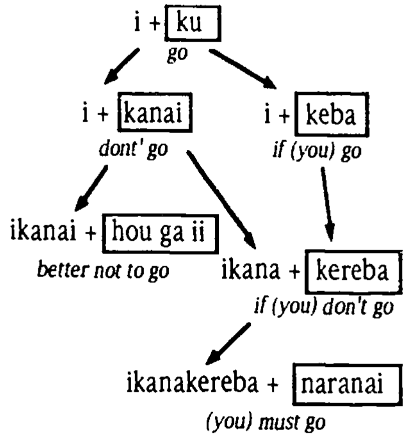


Figure 3. Language Patterns Integration

Language Patterns and Expressions

A *language pattern* is a template representing a general structure for a sentence that includes verb conjugation forms, particles and other grammatical elements. Learning a language pattern implies first understanding the pattern, then producing sentences using it, and finally internalizing the pattern, thereby establishing it as a habit (Alfonso, 1989). Utterances which do not obey tgeneral patterns are called *expressions*. As in the case of language patterns, expressions are also grouped into DSFs where they can be appropriately used. For example, in the case of the DSF "show agreement" we have *hai, wakarimashita* (I understand) and *naruhodou* (I see). Domain knowledge is distributed between dialogue agents, each one representing the different use of language patterns and expressions by Japanese of different gender, age and relative social status.

By providing a pedagogical orientation to our knowledge domain representation, we may relate language patterns through the concept of *integration*. We define integration as the relation between complex patterns which contain one or more basic patterns in their structure (see figure 3). Due to the evidence that language structures are learned in a certain order (Krashen & Terrel, 1983; Carrol, 1986), we consider this integration relation necessary, because it guides the Mediator in the direction of the learner's possible knowledge development.

Student Modeling

Open Student Models

Student models in collaborative learning environments should be open (Self, 1992). For the purpose of establishing a complete communication of goals, needs, and capabilities of the learning group, all student models are freely accessible to the learner at any time with the Learners Performance Communication module. In this way the learners can locate their level with regard to the others (Clement, Vieville & Vilers, 1992). This is relevant for the goal communication process in collaborative learning and curriculum knowledge negotiation (Moyse & Elsom-Cook, 1992; McCalla & Wasson, 1992). In GRACILE, the learner can notice to whom s/he can ask for or provide help while practicing on a specific language pattern or expression.

The Actual and Potential Development Levels

In GRACILE, student models are represented based on the concept of zone of proximal development, defined as the distance between the actual level of development of the learner and her/his potential development level (Vygotsky, 1978). The potential development level represents the knowledge the learner can reach with the assistance of more experienced peers or a teacher. The student model represents the learner's *actual development level* by a set of patterns and expressions which appear in correct sentences constructed by the learner, without any help from the system or other colleagues. This is the knowledge the system believes has been internalized by the learner. On the other hand, the learner's *potential development level* is represented by the set of patterns and expressions used in the construction of correct sentences with the assistance from the student's colleagues or from the system. In Vygotsky's terms, *internalization* occurs when an interpersonal process, at the social level, is transformed into an intrapersonal process, at the individual level (Vygotsky, 1978). We consider that a learner internalizes a language pattern or an expression when this knowledge serves her/his current needs and when it has been already used. We believe that internalization only occurs in the case of knowledge that is within the learner's potential development level (Vygotsky, 1986) and appears in some particular context (Krashen & Terrel, 1983; Carrol, 1986). In addition to the actual and potential development levels representation, the student model contains the set of inappropriately used expressions and patterns, representing the student's needs.

The Mediator

Negotiation of the Learning Activities

Negotiation of learning activities is done between the Mediator and the students following the steps below:
step 1.- The Mediator proposes the subsequent knowledge it believes would be more effectively learned by each member of the group, considering the possible collaboration, based on each student's actual and potential development levels. This learner's group-based knowledge frontier representation, together with the learner's goals, provide the negotiation position of the Mediator.
step 2.- Via the Learners Performance and the Goals and Task Communication and Negotiation modules, learners have free access to the actual development level, needs and goals of other learners, and can locate their own goals with respect to the learning group's. Then, they make explicit their learning goals. The learners also negotiate the role to play in the dialogue, indicating gender, age, and relative social status.
step 3.- Based on the learning group's goals and the learner's group-based knowledge frontier, the Mediator proposes a set of tasks to the student, concerning the construction of sentences for specific DSFs.
step 4.- Learners may accept or refuse the tasks proposed to them.

The Group-based Knowledge Frontier

There is considerable evidence that certain language structures tend to be learned before others (Carrol, 1986) and are acquired in a predictable order (Krashen & Terrel, 1983). Once the student has learned how to use some simple patterns, s/he may then learn more complex patterns which contain the known basic ones more efficiently. Since the learning group members are part of a social entity with similar Japanese language experience, the knowledge that is already used by some learners in the group is necessary for group membership and collaboration and may also be more easily learned by the others. The first task for the Mediator is to determine a representation of that knowledge which can be learned more easily, enhance the learner's progress and provide better collaboration possibilities. As mentioned above, we call this set of patterns and expressions the group-based knowledge frontier (see figure 4). Previous work on ITS proposed modeling of knowledge evolution, as was the case with the term "knowledge frontier" appearing in WURSOR (Goldstein, 1982). In GRACILE, we represent the group-based knowledge frontier by the union of two sets: the learner's potential development level and the complex patterns that can be integrated by those basic patterns which the system believes have already been internalized by the learner. The intersection of these sets is called *candidate knowledge for relevant collaboration*. A degree of usability, popularity, relevance and feasibility is assigned, according to the patterns' integration relations and their use by the group.

Interacting with GRACILE

Learners Performance and Goals Communication

The Learners Performance Communication module of the Mediator allows the student free access to open student models which represent the group members' actual development level, thereby allowing the learners to locate their own level in relation with the others'. Each learner is encouraged to realize who can help her/him or who can be helped. This is an important issue for the process of goals selection and collaboration.

Learners can explicitly express their goals either as *delivery goals* or *content goals* (McCalla & Wasson, 1992). Delivery goals refer to the challenge of constructing an appropriate sentence for a specific DSF, where the learner wants to know and practice how to express her/himself properly. A learner expresses a content goal when s/he wants to know about the usage of a language pattern or an expression that s/he has already noticed, and decides to practice it in a situation where appropriate. The learner can also refuse or accept the task proposed to her/him by the Mediator, during the negotiation of the learning activities of dialogue construction.

Sentence Construction

Learners put their knowledge into practice during the construction of sentences with the aid of a dialogue agent assigned to them and/or the other learners, if necessary. During dialogue construction, the learner works directly with nouns and verbs to construct her/his own sentences. The learning environment is designed to provide the student with Japanese-English dictionaries for nouns, verbs and adjectives written in Kanji and Hiragana. More than 850 Japanese verbs and 80 adjectives can be automatically conjugated by a dialogue agent, whose request input can be a verb or an adjective in English with the selection of the desired conjugation features. The request can be also a conjugated verb or a sentence in Japanese, based on a language pattern or an expression. In this case, the dialogue agent presents the appropriate DSF and the pattern or expression meaning in English, together with the verbal and adjectival conjugation features. Assistance from the dialogue agent is registered into the learner's student model, and represents the learner's current needs concerning the construction and use of language patterns and expressions.

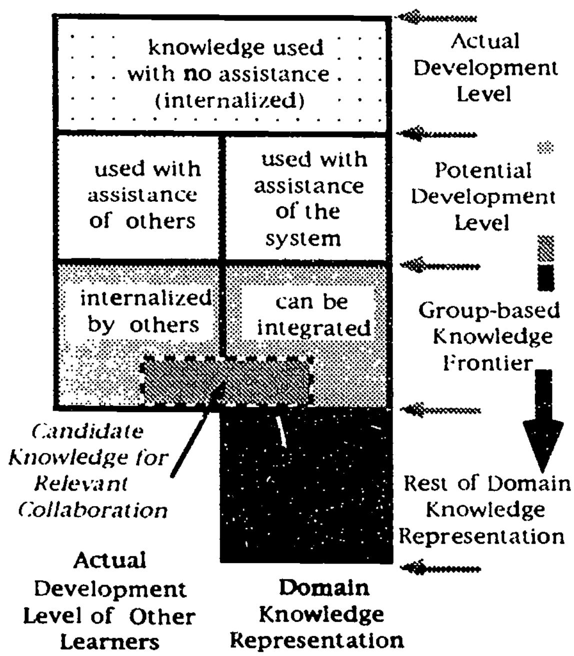


Figure 4. Learner's Knowledge Progress

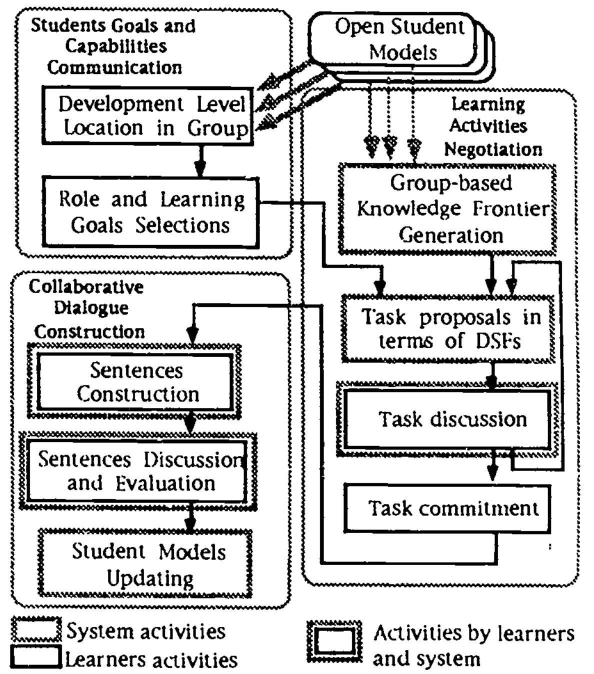


Figure 5. Interacting in GRACILE

Sentence Collaborative Discussion and Evaluation

The knowledge about language patterns, expressions and their appropriate use in a DSF is provided by the dialogue agent, when it receives a request from the learners in order to determine the correctness and the appropriateness of the constructed sentences. The learner may also ask another learner about the appropriate construction and use of a pattern or expression. In this way, the students play both the role of learner as newcomer and as experienced peer. The respective dialogue agent makes an analysis of the constructed sentence, manipulating its knowledge representation of the patterns and expressions according to its age, gender and status. Then the dialogue agent determines whether the sentence is grammatically correct and appropriate to the specific dialogue situation. A constructed sentence, as a part of a shared dialogue, should be accepted by all members and be either discussed or rejected. In the latter case, other learners can assist and propose modifications. The sentence evaluation is expressed in terms of the assistance from other learners or the dialogue agent corresponding to the role the student is playing in the dialogue, which in turn determines if that knowledge has still not been internalized by the learner. After the construction and evaluation of a sentence, the Mediator updates the respective learner's student model. Figure 5 presents the interaction with GRACILE's learning environment in general terms.

Conclusions

We have presented the configuration and the characteristics of GRACILE, a collaborative intelligent learning environment for learning Japanese language patterns and expressions, where learners write sentences together in situations where the domain knowledge is appropriately used. In order to adapt itself to the goals and needs of the learners, the Mediator proposes a set of tasks to the student in terms of dialogue situated functions, based on the knowledge the Mediator considers the student may integrate next and on what it believes has been already internalized by the student and the other members of the learning group. This group-based knowledge frontier is important since it represents the possible collaboration from other peers in the scope of the evolution of the learner's knowledge. We believe that GRACILE is a learning environment that will help Japanese language students develop their reading and writing skills faster, allowing them to become more productive in their communication of Japanese. GRACILE is being developed in Prolog for a network of Macintosh computers.

References

Alfonso, A. (1989). *Japanese Language Patterns: A Structural Approach.* Center of Applied Linguistics, Sophia University , Tokyo.

Carrol, J.B. (1986). Second Language, *Cognition and Instruction*. Dillon, R.F. and Sternberg, R.J. (eds.), Academic Press, London, 83 - 125.

Chanier, T. & Renié, D. (1993). Modelling Language Usage to Enhance Second Language Learning. *Proceedings of AI-ED 93*, AACE, Paul Brna, Stella Ohlsson and Helen Pain, (eds.), 260-272

Chung, J. (1991). Collaborative Learning Strategies: The Design of Instructional Environments for the Emerging New School, *Educational Technology* 31 (12), 5-22.

Clement, D., Vieville, C. & Vilers, P. (1992). An Experiment of Cooperative Learning with Hypercard, in *Computer Assisted Learning, ICCAL'92*, I. Tomek, ed. Springer Verlag, 148-160.

Goldstein, I.P. (1982). The Genetic Graph: A Representation for the Evolution of Procedural Knowledge, in *Intelligent Tutoring Systems*, D. Sleeman and J.S. Brown, (eds.), Academic Press, London, 51-77.

Hayashi, T. & Yano, Y. (1992). Open Structured CAI Systems for Kanji Learning, in *Computer Assisted Learning, ICCAL'92*, I. Tomek, ed. Springer Verlag, 271-282.

Krashen, S.D. & Terrel, T.D.(1983). *The Natural Approach.: Language Acquisition in the Classroom.* Pergamon Press, Oxford.

Kusanagi, Y. (1992). CAI for Teaching Japanese as a Second Language, *Journal of Japanese Language Teaching*, (78), 1- 8 (in Japanese).

McCalla, G. & Wasson, B. (1992). Negotiated Tutoring Needs, Student Modeling and Instructional Planning: Arguments and Experiences from the Perspective of the SCENT Project, in *Knowledge Negotiation*, R. Moyse and M.T. Elsom-Cook (eds.), Academic Press, 41-68.

McManus, M. (1993). The Group Leader Paradigm in an Intelligent Collaborative Learning Environment. *Proceedings of AI-ED 93*, AACE, Paul Brna, Stellan Ohlsson and Helen Pain, (eds.), 249- 256

Moyse, R. & Elsom-Cook, M. (1992). Knowledge Negotiation: An Introduction. In *Knowledge Negotiation*, R. Moyse and M.T. Elsom-Cook (eds.) Academic Press, 1-19.

Pacey, B. (1990). Meaningful Language Learning in the Age of Technology, in *Computers in Education*, A. McDougall and C. Dowling (eds.), Elsevier Science Publishers, 681-686

Self, J. (1992). The Role of Student Models in Learning Environments, in *Proceedings of the IAIA 92* Nagoya, Japan, 103-110.

Vygotsky, L.S. (1978). *Mind in Society: The Development of Higher Psychological Processes*, Harvard University Press, London.

Vygotsky, L.S. (1986). *Language and Thought*, MIT Press.

Hypermedia for Teaching - A European Collaborative Venture

Philip Barker
Interactive Systems Research Group
Human-Computer Interaction Laboratory, School of Computing and Mathematics
University of Teesside, United Kingdom

Antonio Bartolome
Departament de Didactica i Organitzacio Escolar
Divisio de Ciencies de l'Educacio
Universitat de Barcelona, Barcelona, Spain

Abstract: The 'Hypermedia for Teaching' project is a European collaborative venture designed to produce a hypermedia learning package that is published on compact disc read-only-memory (CD-ROM). Two versions of the package are to be developed. One of these is intended to be used on a multimedia personal computer (MPC) while the other is to be used in conjunction with commercially available CD-I (compact disc - interactive) equipment. The MPC version of the package is currently being developed and the CD-I version is being designed. This paper describes the Hypermedia for Teaching project and its current status. It also discusses the logistics and problems of running a large multi-national hypermedia project.

INTRODUCTION

The 'Hypermedia for Teaching' project is a three year research and development project that is funded by the European Community (EC) within its COMETT framework. The project involves a consortium consisting of twelve partners. The partners involved in the project are based in the following countries: Belgium, Denmark, France, Greece, Italy, Portugal, Spain, Sweden and the United Kingdom. The project base is at the University of Barcelona although project meetings are hosted, in turn, by other member countries. The working language for the project is English.

The major objective of the project is to create a collection of hypermedia documents (published on CD-ROM) that can act as an informative and authoritative source of material on hypermedia techniques and their potential uses within a number of different application domains. As well as being of a multi-national nature the project is also required to be multi-lingual in that the materials that are developed must be published (on CD-ROM) in six of the languages of the EC.

The collection of hypermedia documents that is produced will be published on conventional CD-ROM in the first instance -for delivery using a MPC (Jamsa, 1993). Subsequently, they will also be published on optical disc in CD-I format for delivery using a Philips CDI-205 CD Interactive Player. This is a relatively low-cost CD-I delivery platform intended for the consumer market. The reason for publishing the hypermedia documents in two formats is to explore the problems involved in creating hypermedia documents for delivery on multi-platform environments. Currently, no real standards exist -although the EC ESPRIT project OSMOSE (Open Standards for Multimedia Optical Storage Environments) is exploring the problems of defining a common publishing format (DELTA, 1991).

PROJECT DESCRIPTION

This section of the paper briefly describes the Hypermedia for Teaching Project and the progress it has made since its inception in June 1991. The description is organised into four sections: system architecture; implementation notes; CD-ROM prototype production; and problem areas.

System Architecture

Even though the physical implementations of the two CD-ROM discs described above will be different (as will their end-user interfaces) the basic conceptual architecture (and content) of the two CD products will be very similar. The basic common architecture of the two systems is illustrated schematically in figure 1.

This diagram depicts the three-level structure that makes up the complete 'Hypermedia for Teaching' system. The topmost (Contents) level provides the generic control mechanism for the system - through which users gain access to all the other embedded modules and system functions. Beneath this, at the second level, are the various component modules themselves which together make up the overall system. At the third level is the collection of electronic documents that are used as building blocks in order to construct the higher level modules from which the system is built.

LEVEL 1: CONTENTS
[Orientation, User Model, Help, Global Tools, Link to ...]

LEVEL 2: MODULES

CC0 CC1 CC2 CC3 CC4 CC5 CC6

AP1 AP2 AP3 AP4 AP5

LEVEL 3: DOCUMENTS

Document 1 Document 2 Document 3 Document 4

Document 5 Document 6 Document N Document N+1

Figure 1. Basic Structure of the hypermedia system

The Contents module (see Figure 1) provides the only way by which users can enter the application - even though they might leave from any of the lower level modules once they have gained entry to the system. As well as acting as an overall control and coordination facility, the Contents module provides four other basic functions. First, it provides an optional 'orientation' facility for new users of the system. Second, it embeds a simple user model (based upon two dimensions - type of user and the language that he/she speaks); this can be used to tailor subsequent end-user interaction with the lower-level modules. Third, it provides a toolset of generic functions that users might find useful during their interactive sessions with the system. Fourth, it makes available an extensive multimedia help facility (at the generic system level) about such topics as: how to navigate through the system; facts and figures about the CD-ROM itself; details about the project and its participant groups; and so on.

As mentioned above, one of the major functions of the Contents module is to enable users to access any of the second level modules that make up the application. This is accomplished using a mouse-based dialogue by 'clicking' on one of the reactive buttons that make up a two-page menu facility. The modules at this level of the structure (see Figure 1) are organised into two basic categories. First, a collection of 'Common Core' modules; and second, a series of 'Application' modules. The Common Core modules (of which there are seven) each deal (in a generic way) with a particular aspect of hypermedia theory, design or practice. Similarly, each of the Application modules deals (in an application oriented way) with the use of the hypermedia methods within a particular application domain - such as banking, project management, language teaching, and so on. The complete set of Common Core and Application modules used in the project is listed in table 1.

The electronic documents that exist at the bottom level of the structure shown in figure 1 make up the basic building blocks of the overall system. This collection of documents can be interlinked in various ways to form the Common Core and Applications modules that exist at level two. The electronic documents residing at level three can be classified according to two basic dimensions. First, whether they are of a linear or a non-linear format. Second, whether or not they have a simple of compound structure. Simple documents are ones which do not embed any other document whereas compound documents embed one or more other (linear or non-linear) documents within them The embedding of documents one within another can take place in either of two ways - directly or indirectly. Direct embedding involves importing one document into the body of another. Indirect embedding involves dynamic linking to other external documents which are therefore not themselves part of the original document's structure. Obviously, there are advantages and disadvantages to each of these approaches.

Table 1
Component Modules of 'Hypermedia for Teaching'

Designation	Topic Treated	Designation	Topic Treated
CC0	Introduction	AP1	Higher Education
CC1	Basic Concepts	AP2	Company Training
CC2	History of Hypertext	AP3	Foreign Language Learning
CC3	Creating Hypertext	AP4	Project Management
CC4	Navigation	AP5	Banking
CC5	Hypermedia		
CC6	Teaching and Learning		

An example of the way in which embedding can take place is illustrated schematically in Figure 2 which shows the range of different documents involved in the creation of Common Core Module 0 (Introduction). This is one of the largest modules in the 'Hypermedia for Teaching' system and is designed to provide users (having no prior experience of hypermedia) with an introduction to this topic. The main hypermedia document that makes up this module is composed from three basic types of entity: a body; a set of supporting external files (containing text, sound, pictures and video material); and the collection of directly/indirectly embedded documents that it uses.

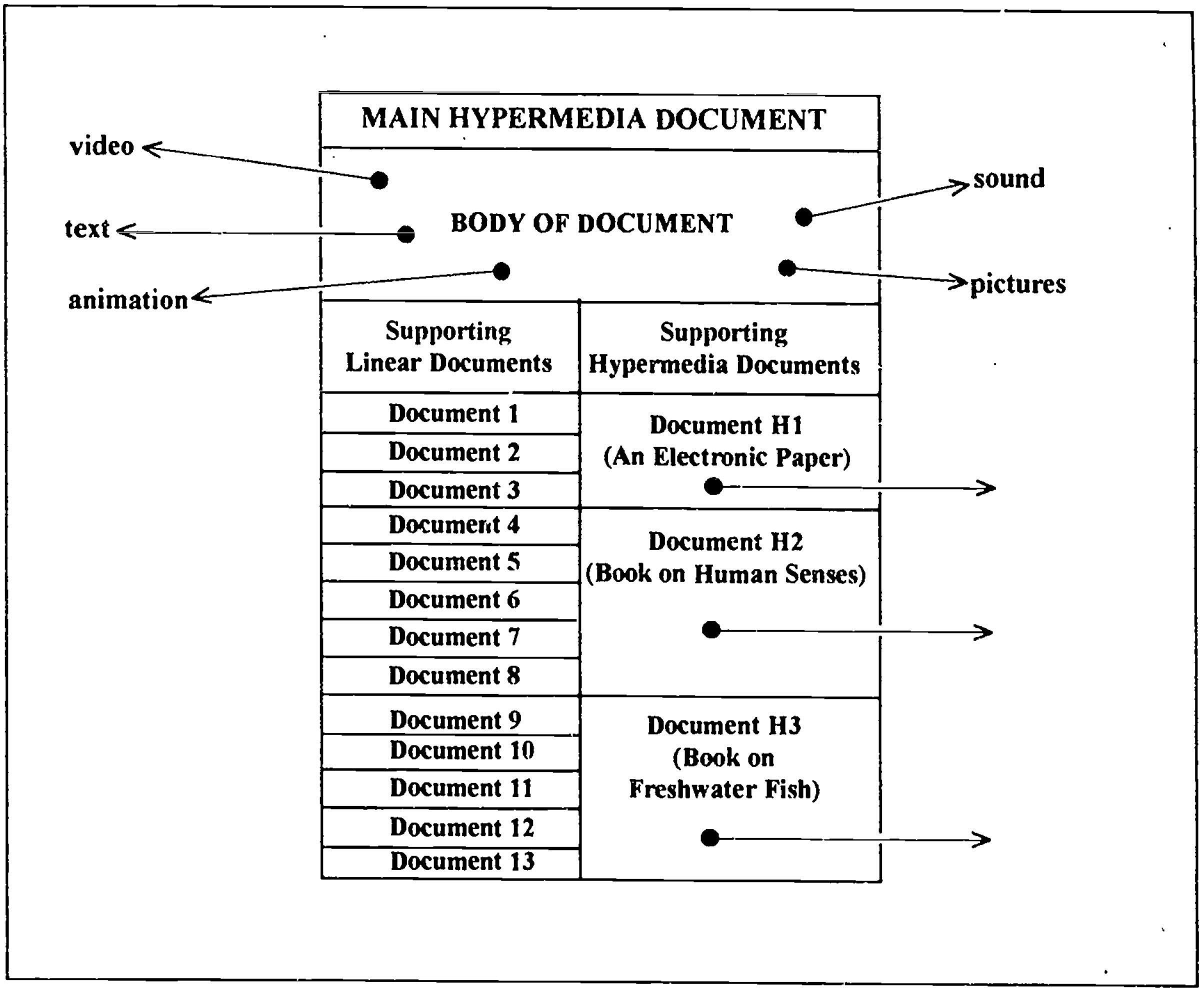


Figure 2. Document types involved in Common Core Module 0

As can be seen from the bottom level of the structure shown in Figure 1, the document body that makes up each hypermedia document is organised in the form of a simple electronic book structure (Barker, 1991). That is, the documents from which the application is composed are regarded as being collections of reactive multimedia and/or hypermedia pages of information upon which are defined various types of operation (such as 'next page', 'previous page', and so on). The operations that are available within any given module will depend upon the types of tool that a particular module needs in order to fulfil its requirements within the overall application.

Implementation Notes

The implementation language choosen for the project was ToolBook (Barker, 1993a). The use of this environment for the implementation of the 'Hypermedia for Teaching' system enabled us to use an object-oriented approach to the design and development of the overall application. We found that simple book and page structures, backgrounds, fields, graphic objects and groups could easily be created using the basic set of tools provided by the ToolBook system. However, some of the more sophisticated object properties and attributes had to be created and controlled using programming techniques. These involved having to generate scripts using the OpenScript programming facility. We also found that most of the simple things that we wanted to do could be easily achieved using the basic ToolBook development environment (Version 1.53) but many of the more complex operations (involving digital sound, animation and motion video) required the use of the multimedia extensions.

The Contents module was implemented as a simple ToolBook book containing just three pages: an introductory page and two menu pages. Each menu page contained two sets of reactive buttons. These were organised into two basic types: local buttons and link buttons. Local buttons could be used to initiate local functions such as page turning, invocation of help facilities, exiting, and so on. In contrast, link buttons were used to invoke the other ToolBook books that existed at the second level of the modular structure illustrated in figure 1. An important component of the Contents module is its multimedia help facility. This is capable of providing help and advice in the form of screen-based text and/or audio narrations. The latter are stored as pre-recorded digital sound files (in WAV format) which are sent (as needed) to the Sound Blaster audio card that is embedded within the host MPC that is being used to deliver the application.

With one exception, all of the Common Core and Application modules were implemented as individual books which were later incorporated and integrated into the complete 'Hypermedia for Teaching' system. In most cases one particular project partner took the responsibility for developing at least one (but sometimes two) of the level two modules. The only exception to this implementation strategy was associated with the Project Management module which, for 'political' reasons was implemented in IconAuthor and subsequently integrated into ToolBook using dynamic linking and object linking and embedding techniques (Jamsa, 1993).

From the point of view of creating hypermedia documents the ToolBook environment undoubtedly provided an exceptionally easy system to use. For example, it was very easy to create new books, backgrounds and pages; and then add text fields, graphical objects, sound effects and reactive buttons to these pages. Buttons were also simple to create (using the button tool) and their reactive properties could then easily be defined by the ToolBook scripting facility. Graphical objects could be created (or imported) and then made reactive by 'pasting' invisible buttons over those areas that were to exhibit reactivity. These reactive areas could then be linked to other pages, to other documents or to other processes. Special effects (visual and audio) could also be created through the design and implementation of appropriate scripts. It was also relatively easy to create reactive words and phrases within text - and then link these up to other pages or objects within the same screen or another book.

Obviously, text handling facilities played an important role with respect to page creation for the hypermedia documents that were used. Invariably, this involved importing text from other applications such as desk-top publishing systems and word-processing packages. In general, no major difficulties were encountered although some minor editing of imported text was usually necessary. We found that two of the most useful features of the ToolBook system (in the context of text handling) were its scrollable text fields and its text searching facilities. The use of scrollable text fields meant that large volumes of hypertext material could be accessed through a relatively small fixed screen area. However, some care had to be taken in following up links from and back to pages containing scrollable fields - in order to avoid loss of context. This problem was overcome by writing a simple script that could 'remember' the context of a scrollable field when one of its embedded hotwords was selected. The ToolBook text searching facility provided a useful mechanism by which end-users could search through hypermedia documents in order to locate particular sections of text that might be of special interest to them.

CD-ROM Prototype Production

The creation of a CD-ROM prototype disc involves a number of fairly well-defined stages (Barker, 1993a). Within this project the four main stages were: local testing and emulation; data transportation; building a disc image on magnetic disc; and transferring the disc image cross to a recordable compact disc using either a single session or a multi-session CD recording unit. Each of these stages is briefly discussed below.

Each project partner was given the responsibility for ensuring that the material submitted for publication was free from errors and coherently organised. In order to realise these objectives it was necessary to use local emulation facilities in order to simulate the behaviour of the modules when committed to compact disc. In many cases this emulation phase involved building the anticipated module structure into re-writable magneto-optical disc storage. In cases where this was not available, hard disc storage had to be used instead.

It was agreed that one project partner (the University of Barcelona) would be responsible for collecting together the modules produced by the different project partners. Various methods of data transportation were therefore used to transfer modules to the Spanish group. In general, three basic techniques were used: tape streamer; exchangeable hard disc units; and conventional 3.5 inch floppy discs containing compressed files that could be decompressed on arrival in Barcelona. Electronic mail facilities were considered but not all partners had access to this type of resource.

At the University of Barcelona all of the materials that were submitted by the project partners had to be extracted from their transportation media and stored in a standard, uniform and appropriately structured way within the host PC that was used to build the CD disc image that would subsequently be transferred to the actual compact disc. Prior to constructing the CD disc image any minor editorial changes (textual, graphical or aesthetic) that were necessary to achieve overall consistency were undertaken.

Once the CD disc image had been created it was transferred across to the CD-ROM recording system (a Philips CDD-521GN Desktop CD-Recorder System) that was attached to the PC that was used for system building. Two basic approaches can be used: single session and multiple session. The first of these involves transferring all of the disc image across to the CD in one single transfer session. The second involves building the CD incrementally over several independent recording sessions. Although our equipment was capable of performing multi-session transfers, we transferred our disc image in one single transfer session.

Problem Areas

Obviously, in a multi-national project such as this one, which involves so many different partners, a number of problem areas will invariably arise. We conclude this section of the paper by summarising some of the difficulties that arose during the project and how they were overcome. For the purpose of the following discussion the problem areas that were encountered are organised into three broad classes: end-user interface design; production issues; and resource implications.

Some of the most interesting problems that we had to resolve were created by end-user interface design issues (Bartolome, 1992). The problems encountered here involved having to decide upon: when to use icons and when to use textual descriptions for control functions; the text fonts and point sizes to be used; the types, style, size and positions of buttons; what tools and facilities should be provided for end-users; when and where to use menu entries and buttons (in situations where a choice was available); and the levels of help that should be provided by the system. All of these problems were eventually resolved after much (often heated) discussion and debate.

Overall, the production of scripts, modules and the CD-ROM proceeded fairly smoothly. However, a number of obstacles did have to be overcome. Three of the most important of these related to: consistency of treatment (with respect to the level and scope of subject matter); consistency of style (that is, ensuring that different modules produced by different partners all had the same style of presentation and a consistent end-user interface); and programming efficiency - partners varied widely with respect to their knowledge of and expertise with ToolBook and OpenScript. Again, these problems were not too difficult to overcome as a result of the free interchange of technical programming knowledge, the creation of an editorial advisory team (whose responsibility was to check the quality of scripts prior to their implementation) and the appointment of a quality assurance officer (who was based at the University of Porto in Portugal).

As is the case with virtually all technology-based projects, the problems arising as a result of limited resources were probably some of the most difficult to overcome. There was a fixed amount of financial resource (in terms of ECUs) available for the production of the 'Hypermedia for Teaching' discs. This had to cover the costs of equipment, development, translation between European languages, travel and project meetings. Fixed sums were therefore allocated to each one of these expenditure headings. Undoubtedly, the mechanism for dividing up the budget available for module development created one of the most difficult problems to sort out. In the end it was decided that all modules should be allocated equal funding. Obviously, in some cases this created problems with respect to the 'natural closure' of a module and hence its overall size.

Obviously, some problems still remain to be resolved. Two of the most important of these relate to the translation of the overall system into the six selected European languages and the logistics of creating and distributing a multi-lingual system such as this. These problems will be resolved in due course. Despite the problems that have been encountered in this project, overall, it has made good progress and will undoubtedly be successfully completed.

CONCLUSION

Electronic documents are playing an increasingly important role as a mechanism of communication within a wide range of different contexts such as business, commerce, education, consumer services, and so on. In the past most electronic documents contained just text or text augmented with simple images. However, the advent of multimedia and hypermedia technologies (involving the combined use of text, sound, animation and motion video) now means that many more different types of electronic document can be created (Martin, 1990; Barker, 1993b; Richards, 1993). Hypermedia documents are particularly interesting because of the potentially high communication bandwidth that they are able to make available. Of course, such documents are much more difficult to prepare and disseminate (compared with conventional electronic documents) because they can involve many more communication modalities.

Of course, a major problem associated with the creation of hypermedia documents is the problem of closure. That is, trying to draw reasonable and relatively natural boundaries around the content domain of such documents. Because of the principle of inter-relatedness (which states that, in theory, 'everything is related to everything else'), creating natural closure in hypermedia documents is much more difficult than doing so in conventional documents. It is therefore very difficult to answer the basic question 'Is a hypermedia document ever complete?'.

Another important problem associated with the use and sharing of hypermedia documents (particularly, when they are intended for international use) is the difficulties associated with their translation into other languages. We have found that considerable difficulties can arise - even when they are all produced in a common base language (such as English) and then translated into other target languages.

Our experience in this project with MPC and CD-I design and production leads us to believe that a significant problem that will have to be faced in the near future is the problem of agreeing upon a common publication format and delivery platform for hypermedia documents. At present no real internationally agreed upon standards exist in this area -although some groups are actively working on this issue. However, until such standards have been decided upon and implemented it will prove very difficult to exchange hypermedia documents and deliver them on different delivery platforms unless some form of mark-up language (like SGML) is used.

REFERENCES

Barker, P.G., (1991). Electronic Books, Special Edition of *Educational and Training Technology International*, 28(4), 269-368.

Barker, P.G., (1993a). *Exploring Hypermedia*, Kogan Page, London.

Barker, P.G., (1993b). Authoring Hypermedia Documents, *Journal of Document and Text Management*, in press.

Bartolome, A.R., (1992). Interactive Levels and Cognitive Styles: the Problem of Design, Learning Resources Journal, 8(3), 63-69.

DELTA, (1991). *Developing European Learning through Technological Advance (DELTA) - Exploratory Action - Final Technical Report (DE2205)*, CEC, DG XIII, B-1049 Brussels, Belgium.

Jamsa, K.A., (1993). *Instant Multimedia for Windows 3.1*, John Wiley & Sons, New York, NY, USA.

Martin, J., (1990). *Hyperdocuments and How to Create Them*, Prentice-Hall, Englewood Cliffs, NJ, USA.

Richards, S.R., (1993). *Human-Computer Interfaces to Electronic Books*, PhD Thesis, Interactive Systems Research Group, School of Computing and Mathematics, University of Teesside, Cleveland, UK.

Hypermedia Tutoring System : Towards an Architecture Dedicated to the WEB Specification

T. BELTRAN, A. PÉNINOU
A.P.I. Lab., Université Paul Sabatier - IUT A
50 chemin des maraîchers, 31062 Toulouse, France

Abstract: In this paper, we present the first stage of the HyTuS project (Hypermedia Tutoring System). This project originated by the *Alliance Française* Institute is a cooperation between University and professional partners (*Alliance Française* is a training institute supplying French courses in France (for foreigners) and abroad. It is composed of more than 2000 centres all over the world). The aim of the project is to provide a hypermedia CD-ROM-based environment for learning French according to the *Alliance Française* educational programs. In this paper, we have studied the problem related with its distribution. The aim is to provide the learners with specific views (WEBs) of the CD-ROM database. Two aspects are considered : the first one, purely pedagogical, tackles the need to focus a learning session to a restricted number of topics. The second one is a technical problem : some training centres are still based on traditional teaching methods and are not well-fitted out with high quality computers. Consequently, some of them cannot put into practice CD-ROM-based training sessions. The solution is to provide them with a magnetic disk containing particular sub-sets of the educational hypermedia CD-ROM version.

Aims and framework of the HyTuS project

The first stage of the project was initiated by a profitable team-work in which pedagogues and teachers' needs and dreams were confronted to computer scientists' reality (costs, calendars, ...).

It was decided to focus the prototype to the second level morpho-syntax aspects of the *Alliance Française* institute learning program. Taking into account the diversity of learners (age, nationality, curriculum, ...) was one of the major French Institute requirements. Fortunately, this problem was yet well resolved (at least theoretically) in A.P.I. works (Beltran, 1991; Beltran, 1992; Beltran, 1993), ...

Defining Educational Hypermedia sessions

The hypermedia is designed to be adapted to several pedagogical aims and contexts (several strategies related with the second level of the learning program are implemented). The CD-ROM covers several months of traditional training, but learning sessions have to be short and focussed on topics defined by both learners and instructors. Consequently, we have to manage this fragmentation, providing the learner with sharply defined views of the knowledge base (or 'WEBs', according to the hypermedia literature).

• Virtual (or logical) partitions : a pedagogical requirement

The WEB structuring of knowledge is a classical mean which aims to avoid getting lost in a hypermedia document, particularly when they are used for CAL purposes (Richards & Mühlhaüser, 1993). But, the definition of navigation strategies and user models is not enough in the CAI context of our architecture (Hypermedia-based tutoring system). The web definition must be still stronger because a HyTuS web has to be organized according to particular pedagogical aims (not only navigation contexts) : revision of concepts, overview, detailed presentation, exercises, knowledge checking, ... Each pedagogical strategy implies a particular behaviour of the supervision system, that is to say a particular web management. On the other hand, new constraints impose the creation of physical partitions.

• Physical partitions : a technical and organizational requirement

The CD-ROM solution seems to be the best support for data storage and distribution. However, some *Alliance Française* training centres are isolated and poorly equiped (with personal computers of several capabilities). So the tele-teaching architectures proposed in numerous works (TeleTeaching, 1993; Applica, 1993), (network-based and often requiring expensive devices) are unsuited for our problem. This has led us to study the distribution of hypermedia webs (particular views) using more traditional supports, such as removable hard disks (e.g. SyQuest) or floppy disk packages. Later on, we shall describe the web extraction mechanism.

The WEB specification (see figure 1)

The model we propose (Beltran, 1993), constitutes an extension of a widely accepted hypertext reference model : the DEXTER model (Halaz & Swartz, 1990; Safe, 1990). The educational hypermedia CD-ROM is composed of hypermedia knowledge (information, links, ...) plus, the textual representation (DEXTER formal description of the hypermedia document : (a) in fig.1). This description, or a copy (a'), is used by the 'WEB specification tool' in order to define the WEB corresponding to the parameters stated by the instructor (b) : learner profile, pedagogical aims, characteristics of the target computer(s), ... A dialogue with the instructor allows him to refine the WEB to be produced. Next, from both the WEB specification document (c) and the hypermedia database, the extraction tool builds a physical partition, satisfactory from both pedagogical and technical points of view (d).

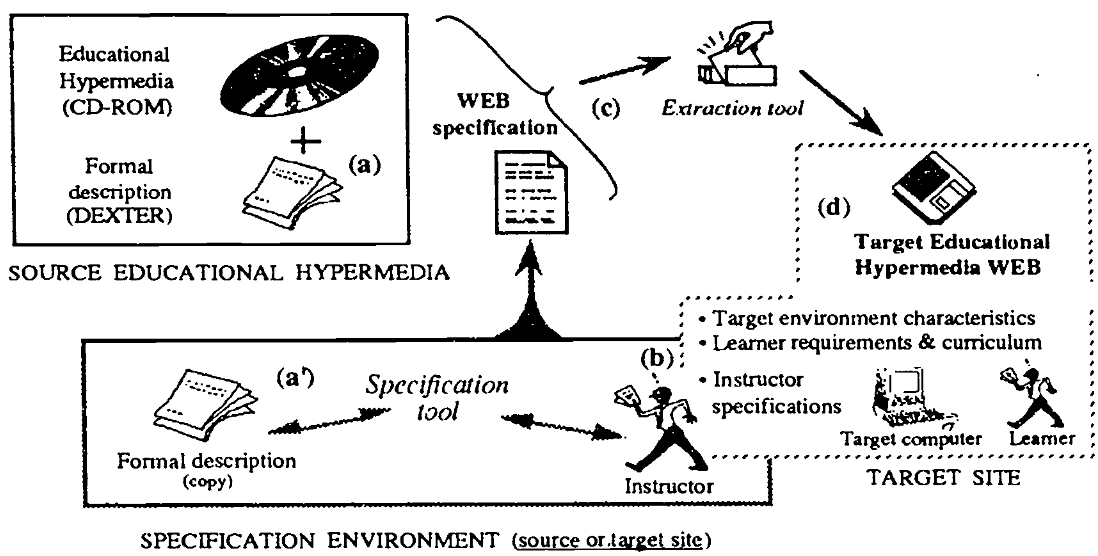


Figure 1. Specifying an Educational hypermedia WEB

Three possible configurations

• Each computer of the training centre is connected to a CD-ROM device (directly or by network): at the beginning of a learning session, the learner chooses among a set of WEB specifications prepared by the instructor. The WEB is automatically computed by the extraction tool at navigation time.
• The majority of training computers is not connected to a CD-ROM device : in spite of this, the instructor can specify the WEB (only the formal representation is needed). Next, if a computer of the centre can read a CD-ROM (local or remote device), the WEB is built and stored on a magnetic support. To be used by the learners, the WEB should be installed on the target computers.
• In the case where no CD-ROM device is available, the instructor sends the web specification document to a centre having CD-ROM devices. The WEB is built and returned to the instructor.
This process allows poorly equiped centres to provide the learners with adapted learning programs (e.g. presentation of the first month topic courses, revision of sequence 1, exercises and complements of modules 4 and 5, ...).

The on-going HyTuS prototype : the Hyperdocument structure

Organization of knowledge (figure 2)

Structuring knowledge in Hypermedia sequences, composed of several modules allows, at the upper level a topical navigation. Moreover, this organization suits the WEB specification.

The learning program is composed of several sequences; each of them is focussed on a particular problem (such as the negation : '*ne .. pas*') and contains modules which treat a particular aspect of the sequence main subject (for example : a module to learn sentences including '*ne pas...*'; another where '*ne*' and '*pas*' are separated with only a verb; ...). Several types of multimedia Hypermedia-organized exercises are proposed; complements can be consulted.

According to the strategy stated by the instructor or the learner, the hypergraph of the module is dynamically computed. The evaluation of the learner interaction within a module depends also on the HyTuS strategy. Links to or from other modules of the same sequence are managed by the HyTuS pilot.

The Hypermedia Tutoring System allows the user (instructor or learner) to decide which strategy will be used during the session. Several parameters can modify the supervision of the learner interaction with a module, the paths within a module, the linkage of sequences, ... A set of relevant supervision parameters has been decided by the pedagogues of the *Alliance Française* Institute.

A tool for HyTuS behavior specification is now being developed and allows to set the following supervision settings:

Notation	**Access to complements**	**Module browsing**
• 20 points if answer is found, 0 if not • Depending on the number of possible answers	• Always • Only if the point average > 'x' • Never	• Overview (x% of exercises) • Complete • Learner choice

Exit module (link to another module)	**Answer analysis**	**Exit sequence** (to another sequence or end)
• At end • If point average > 'x' and 'y' % of exercises are done	• After each exercise • After each module • After each sequence	• At end • If point av. > 'x' and 'y' % of modules • According to learner errors

Error treatment (link between sequences)	**Sequence and module calling**	**Sequence and module priority**
• Do not take into account • Take into account but continue • Treat immediately & go back • Treat immediately & don't go back	• According to priority list(s) • According to learner errors	• Instructor list • Learner list • Predefined lists depending on the learner language

Taking into account the CAI components of the educational hyperdocument

In this purpose, the MACT model (Péninou & Gouardères 1993a; Péninou & Gouardères 1993b), designed to knowledge base management in ICAI systems, seems particularly suited. This model is based on the notion of pedagogical primitive (also refereed in Woolf, 1991; Van Marcke, 1992; Major & Reichgelt, 1992). MACT defines four components:

• an Object base representing the system reasoning context (curriculum, available pedagogical styles, learner interaction, ...).

• a Task base representing high level functions of the system. Each task represents a pedagogical primitive and takes into account a system pedagogical function or a system pedagogical objective. A task defines the set of shared objects and describes the identifiers of needed tasks, considering a specific context. Several agents are associated to a task.

• an Agent base, in which each agent is designed to perform its associated task in a specific context. This context is determined by 'pre' and 'post' condition attributes defined for each agent.

• a run-time engine which runs MACT specifications, manages the object base sharing and the selection of agents in order to realize a task (by checking 'pre' conditions and selecting agents). This mechanism can be compared with the functioning of the GTE system (Generic Tutoring Environment, (Van Marcke, 1992)).

This model, adapted to our needs, allows the definition of pedagogical contexts within an educational hypermedia document. It is also a good support for the extraction of coherent pedagogical WEBs. Let us now present the external structure of the HyTuS Hyperdocument being developed.

The Task / Agent internal structure of the Hypermedia Tutoring System

The Hypermedia specification (based on the DEXTER model, extended using the MACT formalism) constitutes a standard description of both hypermedia information and supervision components. The pedagogical primitives tasks allow the representation of pedagogical aims related to a specific use of the hyperdocument.

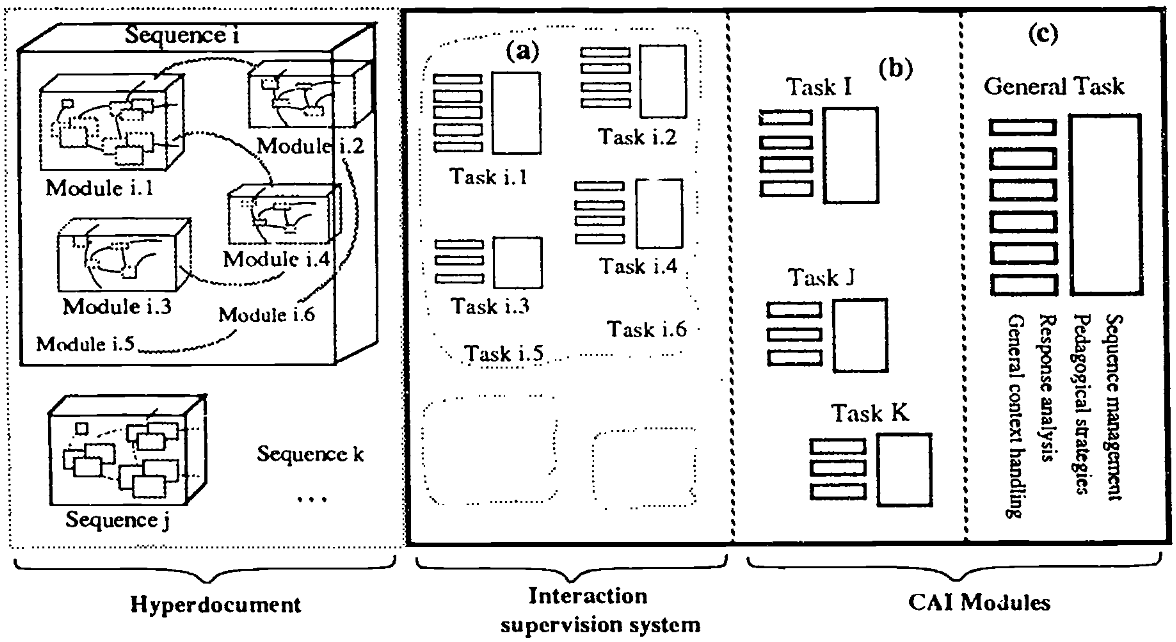


Figure 2. The tasks / agents internal structure

This representation involves the three levels of the educational hypermedia : the hyperdocument, the supervision system and the CAI components (see figure 2).

• A task, named 'Module Task' is linked to each module of the hyperdocument (a). Agents, named 'Module Agents' attached to this task describe the module management in a specific context. Each agent is owner of a list containing the Static Links (classical links: SL) and another containing the Pedagogical Links (supervised dynamic links : PL) defined for the module. These two lists allow the system to know which information is needed when the module is activated, according to particular supervision parameters (exercise notation, exit module parameters, access to complements, module browsing, answer analysis, error treatment).

• A task, named 'Sequence Task' is associated to each sequence of the hyperdocument (b). In the same way, agents named 'Sequence Agents' are associated to this task. A list of 'Needed Modules' (NM), addressing the modules able to be activated is associated to each Sequence Agent. These agents define the sequence management with regard to the parameters stated by the instructor, i.e. the system behaviour settings (access to complements, sequence browsing, exit sequence parameters, answer analysis and error treatment).

• Finally, the HyTuS pilot is represented by the 'General Task' (c), its agents, named Pilot Agents define the parameters related to the supervision context : error treatment, sequence or module priority and calling, ... A list of 'Needed Sequences' (NS) addressing the sequences able to be activated is associated to each Pilot Agent.

The object base contains : • data related to the supervision parameters, • objects shared by both the Pilots Agents and the user interface management system (event-based system).

The agents connected to each task are in charge of one or several combinations of the supervision parameters. The activation of an agent (controlled by the attributes named 'pre' and 'post' conditions) depends on the values of both • the hypermedia / supervision system shared objects and • the supervision parameters.

Specifying WEBS

A Web specification consists in delimiting particular hypermedia knowledge with regard to : • the possibility to install the Web on a target computer, • the satisfaction of the constraints stated by the instructor related both to the information content and the interaction supervision. Extractions are made thanks to queries based on the DEXTER representation extended with the pedagogical oriented MACT overlay. We propose a step by step approach, resumed by the figure 3.

Topic / Subtopic Selection (DW : Domain Web) From a topic browser and a table of contents presenting the hypermedia units (sequences and modules), a first set of queries defines a Domain Web (DW). This DW is only content satisfactory in regard of the instructor specifications.

Supervision Parameters Setting & Task / Agents Extraction (PW : Pedagogical Web) Each DW supervision task is analysed : according to the supervision parameters, only relevant agents (for pedagogical management) are selected (module agents, sequence agents and pilot agents having relevant activation 'pre' conditions). From the selected Module agents and their lists of links (SL and PL lists), a PW Web is generated.

Topic / Pedagogical Completeness checking (DPW : Domain Pedagogical Web) DW and PW do not always strictly overlap (PW units may not belong to DW, or reciprocaly). In order to obtain the resulting DPW, a compromise between DW and PW must be found (feed-back to the two previous steps). At this stage, selected domain information is linked to the selected control agents.

Technological Requirements & Improvements (DPTW : Domain Pedagogical Technological Web) From the DPW and the extended DEXTER representation, the Web size estimation is made and compared with the target computer capabilities. Adaptations can be achieved (modifying pictures and movies quality, sound definition, ...) in order to fit the computer characteristics. The resulting WEB is analysed from a technological point of view.

When the Specification DPTW matches a Good pedagogical / technological Compromise, the web is found. Else, if DPTW characteristics do not match the instructor requirements, a feed-back to the previous steps is needed (the resulting web will be truncated or extended).

Satisfaction Criteria The final Web size can be reduced according to several points of view :
- only few sequences are needed (those refereed in the selected pilot agents NS lists),
- only few modules are chosen (those refereed in the selected sequence agents NM lists),
- concerning the selected modules, only few agents are needed, so, only few units of hypermedia information (SL and PL lists of the module agents) are associated to the final web,
- size of information can be tailored to technological requirements.

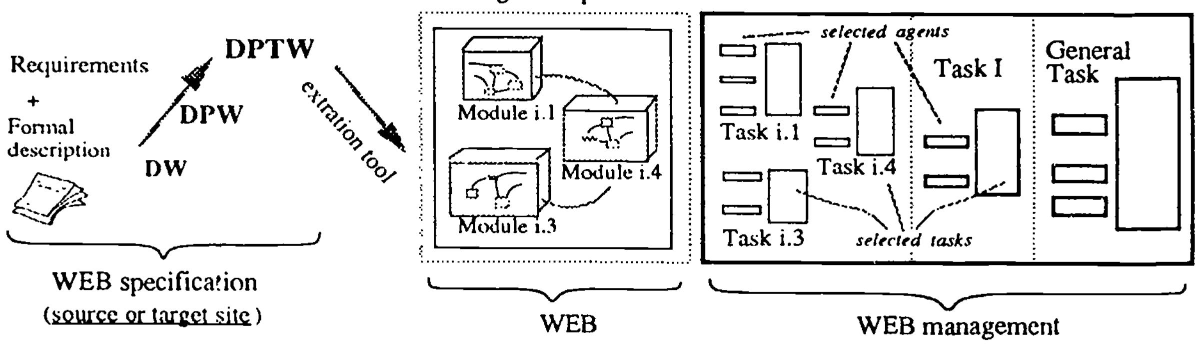


Figure 3. DPTW specification and physical WEB production

Extraction tool

The extraction is made from the DPTW specification ; hypermedia units and their supervision tasks have to be isolated :
- Information data is copied from the CD-ROM database,
- Links between the hypermedia units are copied (DW),
- Modules, Sequences and Pilot Agents are reorganized (DPW),
- Needed pedagogical and static links are copied (SL and PL lists), related information is stored,
- The object base is reduced to contain the only needed supervision parameters, and the necessary hypermedia information addressed by the selected agents.

Then the program, based on the target computer educational hypermedia management system and the MACT engine is produced.

Results and Prospects

In this paper, we have anticipated the second project stage, studying the theoretical and practical problems related to the educational hypermedia distribution. The WEB specification and extraction mechanisms are presented as a necessary solution from a technical point of view, but we also have focussed on the very interesting aspects involved from the pedagogical point of view. The first WEB specification and extraction experiments from an extended DEXTER formalism have been satisfying, although they have underlined the necessity to have heuristics allowing to help the instructor during the specification process. Other heuristics dedicated to the WEB optimization regarding to the target site constraints (e.g. screen or memory capabilities) are needed. In conclusion, from this first practical experience, we expect to make the theoretical model evolve, and we hope to draw a framework for optimizing the educational hyperdocument WEB specification... This study can also stands as a basis for structuring CAI/ICAI systems from which satisfactory and coherent parts can be extracted and run. But a great part of work is still to be done.

References

Applica (1993). Proceedings of Applica'93 / TelePresence, First International conference in Technologies and Theories for human cooperation. Lille, France, 22-24 March 93.

Beltran T. (1991). *Hypermedia and Learning: an architecture for hybrid systems* International Conf. CALISCE'91, Computer Aided Learning and Instruction in Science and Engineering, Lausanne, Swiss, 9-11 Sept. 91, pp. 167-176.

Beltran T. (1992). *Hypertext, DataBases and Computer Aided Instruction : where is the match ?*, Proceedings of the 3rd International conference 'Databases and Expert Systems Applications' (DEXA'92), Valence, Spain, 2-4 Sept. 1992, pp. 215-220.

Beltran T. & Péninou A. (1992). *An Object Oriented Approach to produce hypermedia Educational Software*, 4th International Conf. 'Computers and Learning': ICCAL'92, Wolfville, Nova Scotia, Canada, 17-20 June 1992, pp.111-123.

Beltran T. (1993). *Educational Hypermedia : from theory to practice*, World Conference on Multimedia and Hypermedia for Learning (ED-MEDIA'93), Orlando, Floride, 23-26 June 1993, pp.68-75.

Halasz F. & Swartz M. (1990). *The Dexter hypertext reference model*, NIST'90, Hypertext Standardization Workshop, 16-18 July 1993.

Major N., Reichgelt H (1992). *COCA : a Shell for intelligent tutoring systems*, International Conference 'Intelligent Tutoring Systems' ITS'92, Montréal, Canada, Lecture Notes in Computer Science 608, Springer-Verlag, June 1992.

Péninou A., Gouardères G. (1993a). *Structuring Knowledge for ICAI systems : MACT model's contribution*, 13th international conference 'Artificial Intelligence, Expert Systems, Natural Language' AVIGNON'93, Avignon, France, May 1993.

Péninou A., Gouardères G. (1993b). *A Functional Architecture of Knowledge to structure Intelligent Tutoring Systems*, International Conf. 'Computer Based Learning in Science' CBLIS'93, Vienna, Austria, Dec. 1993.

Richards M., Mühlhaüser M. (1993). *PreSript : A typing approach to the Construction and Traversal of Hypertexts*, World Conference on Multimedia and Hypermedia for Learning (ED-MEDIA'93), Orlando, Floride, 23-26 June 1993, pp.436-443.

Safe (1990). Hatzopoulos M., Dimopoulou K., Gouscos D., Spiliopoulou M. Vassiliakis C., Vasirgiannis M.: *HYP/20 Final report on hypermedia databases*, DELTA project D1014: Standard Authoring Facility Environment (SAFE), 20 nov. 90.

TeleTeaching (1993). Proceedings of the IFIP TC3 Third TeleTeaching conference, Trondheim, Norway, 20-25 Aug. 93, Davies G. & Samways B. editors, ISSN 0926-5473.

Van Marcke K. (1992). *Instructional Expertise*, International Conference 'Intelligent Tutoring Systems' ITS'92, Montréal, Canada, Lecture Notes in Computer Science 608, Springer-Verlag, June 1992.

Woolf B.P. (1991). *Representing, Acquiring, and Reasoning About Tutoring Knowledge*, Intelligent Tutoring Systems : Evolutions in design, Edited by Burns H., Parlett J.W., Luckhardt Redfield C., Lawrence Erlbraum Associates Publishers, Hillsdale, New Jersey, 1991.

Accessing distributed multimedia documents for instructional use

J. BOURDEAU, I. DELATEJA, G. PAQUETTE
Licef, Télé-université, P.O.Box 5250, Station C,
Montréal (Québec), H2X 3M4 Canada
Email: bourdeau@teluq.uquebec.ca

A. FRIHIDA, J. GECSEI,
Laboratoire multimedia, Université de Montréal
CP 6128 Montréal (Québec) H3C 3J7 Canada
Email: Gecsei@iro.umontreal.ca

Abstract: Students, teachers and authors need to access multimedia documents over heterogenous networks. Quality of Service (QoS) is a major consideration for instructional use. A Distributed Multimedia Database (DMD) was developed to answer these needs, allowing negotiation of QoS. The DMD has been connected to an Integrated Learning Environment, and a scenario has been designed for Computer Supported Collaborative Learning activities using DMD. The Distributed Jigsaw scenario was implemented and tested. Accessing multimedia documents over heterogenous networks can benefit students and authors. Facilities for QoS negotiation is critical for instructional use.

Students, teachers and courseware authors need to access multimedia documents that are located on different sites at a distance. How can they have access to these documents over heterogenous networks? What are the limitations and potential of different network configurations for instructional use? To answer these questions, a simple version of a distributed multimedia database (DMD) was developed providing remote access to audio, video, image and textual materials distributed over a number of sites. This DMD works over heterogeneous networks and it includes facilities for negotiating the Quality of Service. It has been tested in order to establish its praticability for instructional use. The connection with an Integrated Learning System has been done for direct access in a situation involving the creation of multimedia documents in an instructional context. A more elaborate scenario involving the use of DMD in a collaborative learning situation, called Distributed Jigsaw, has been implemented.

Needs for Instructional Use

Accessing multimedia documents distributed on different sites, among which some are at a distance, is a situation faced by students, teachers, and courseware authors. Students have as an assignment to make a search and report, or to create a multimedia document. Teachers look for material to present to their students, to prepare an assignment or an exam. Courseware authors need to screen large amounts of multimedia documents, select some of them, and edit them to produce instructional material.

Educational institutions, schools, colleges and universities, and often industries, do not have access to sophisticated infrastructures; they need technological solutions that respond to their types of activities, and that are flexible enough to take into account minimal infrastructures as well as more performing ones. The solution described below includes the design of a distributed multimedia database open to heterogenous network configurations; it states the potential and the limits for instructional use.

Database design and network configuration

Since no distributed mul''media database management system is readily available for the Apple Macintosh environment, a simple distributed multimedia database (DMD) was developed and implemented as a distributed multimedia file system augmented by some search capabilities. DMD provides for basic remote acces to multimedia files as required for instructional use, and can also be used as a testbed for experimenting with QoS problems.

Network context

High-speed communication facilities suitable for distributed multimedia applications are gradually becoming available. However, due to technical and budgetary constraints, networks currently available for experimenting are *heterogeneous,* consisting of different transmission lines and protocols. This is particularly true in situations where the network topology is fixed in advance. The direct consequence is that the quality of connection between different pairs of workstations is generally different; this in turn affects the conditions of access to remote multimedia files distributed over the workstations.

Two sites at the Université de Montréal and Téléuniversité campuses were interconnected, each site having a number of workstations. The network configuration consists of a low-speed line between the campuses (modems @14.4 Kbaud/sec, yielding speeds up to 32 Kbits/sec), connecting two LANs (ethernet), one on each campus.

Database

DMD is built upon two components: Apple Remote Access (ARA), and the Alias mechanism of OS 7.1. ARA permits access to the directories of a remote machine through a telephone line. Each workstation holds in local mass storage a number of audio and video files (in Quicktime format), as well as textual and PICT files. The purpose of DMD is to give access to all files in the system to all workstations. Ideally, access should be *transparent;* that is, when accessing a file the user should not know whether it resides in the workstation's storage or if it has to be imported from a remote site. All files in DMD are maintained in a *global directory (GD)* ; each workstation holds a copy of the GD (Fig. 1). In addition, each workstation has a *local directory* of DMD files residing in that workstation. A file which does not reside in a workstation WS is represented as an *alias* (i.e. a pointer to the original file containing its filename, volume and network adress) in WS's version of GD. A semi-automatic update procedure is used to keep the GD updated when files are added or deleted at workstations. The update takes place every time the application program is entered. For each file F, the GD entry contains the following attributes: a) name of the home workstation and the alias of the local name of F; b) title; c) type, i.e video, audio, image, or text; d) capacity and length, for audio, video, and language when applicable; e) owner; f) descriptors or keywords featuring the contents.

Quality of service

The efficient handling of time-dependent multimedia information such as audio and video in distributed systems requires considerable network and computing resources. The term Quality of Service (QoS) refers collectively to the specifications of these resources for a given type of service, and to the mechanisms through which the resources can be allocated. For example, to support video connections, throughput must be maintained continuously for acceptable video playback, and jitter (the variation of transit delay) must not exceed a certain maximum for digitized audio to be intelligible. Depending on the encoding used, audio and video data can tolerate a certain percentage of packet loss and bit errors. Some of these QoS requirements are contadictory and difficult to achieve simultaneously. For example, error control through retransmission causes additional transit delays and increased jitter. In addition to the communication of time-dependent information, control information sent between system components, such as real-time messages, is also subject to QoS constraints. In fact, QoS can be thought of as the set of *all* parameters characterizing a given distributed environment, including the range of available QoS parameters, and the facilities to negotiate them. For these reasons, QoS is becoming a central issue in distributed multimedia system design, as in the proposed Quality of Service Architecture (Campbell & al., 1993) which offers an integrated framework for QoS specification and resource control over all architectural layers of a system. QoS is also being incorporated in high-speed communication protocols such as XTP (Miloucheva & Rebensburg, 1993).

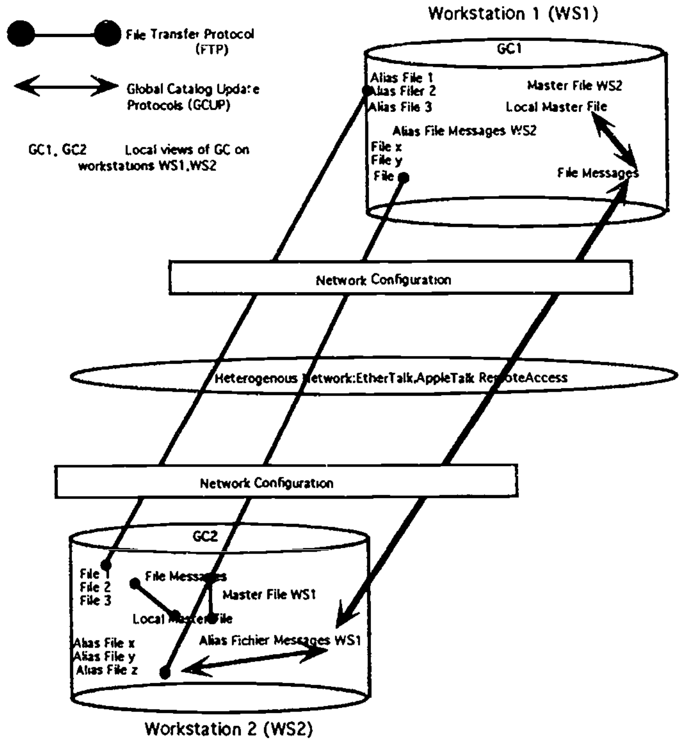


Figure 1. Structure of the Distributed Multimedia Database

A fundamental aspect of dealing with QoS is the set of interfaces through which desired levels of QoS can be requested and negotiated. They collectively allow end-to-end QoS negotiation from the user level down to the network layer. Traditionally, QoS was applied to certain features of network services, as in the transport layer, without any possible control by users. However, user control seems desirable in multimedia applications where the QoS has direct effects on the user's perception of the application, and where the system may not be able to correctly determine the user's preferences. For example, when multimedia documents are to be retrieved from remote sites, the image and video components might be displayed at high resolution, but at the price of some delay and additional cost. Alternatively, lower resolution and/or smaller displays size would be available faster and at lower cost. The DMD interface provides user control depending on the available network speed, as discussed in the next section.

DMD interface

The file system is implemented on Macintoshes in HyperCard, and the user interface consists of the following main component windows: a) a global catalog named GC, showing the list of all files available in DMD; b) attributes of the selected files; c) file display showing the retrieved file; d) audio and video control panel.

Two search modes are available, corresponding to two types of interface screen arrangements, not including another interface used for entering new files and modifying the attributes of the existing files.

The Direct search mode allows the user to make a selection from the GC. The attributes of the selected file are automatically displayed for inspection, but the file is not requested for access.

The Attribute search mode allows formulation of Boolean queries with the values of some file attributes, e.g. type, language, content descriptors. As the result of a query, a subset of the GC is generated with the files satisfying the query. A selection can then be made, as in the Direct search mode.

Accessing the selected file is done by calling the program module *network component* (NC), whose purpose is to maintain information on the underlying communication network. In the present case, it consists of a (fixed) matrix showing the available data rates between each pair of workstations in the system. In more complex situations, NC would contain additional QoS data, which might be variable in time. Conditions of accessing (e.g. waiting time) the desired file are determined through combined information from NC and GC, with three possible options:

1. The file is available locally on the user's workstation WS.
2. The file is on a workstation linked to WS through ethernet.
3. The file is on a workstation linked to WS through a 14.4 kbaud modem.

The negotiation process is the following: the user is first informed about the address of the file and the expected waiting time; she is then offered a choice as to the possible actions. In the case of video files, the choices include to view the video file directly from the source (in cases 1 and 2), to download the file to WS (cases 1, 2, 3), or to download pre-selected sections of the file, or its iconic or textual representation (case 3). Downloading entire video and audio files in Quicktime format is a very time-consuming process in case 3; that can be done off-line. This situation may rapidly change with the availability of new codecs. With more powerful equipment, real time viewing of video material through low speed telephone lines becomes possible.

Testing the system for instructional use

A series of tests has been conducted in order to establish quality of service by the DMD and its praticability for using it in learning or authoring situations. Results show that:

1) search and browsing through the DMD are effective and offer good quality,

2) importation of MM documents requires long delays that make it necessary for users to plan either overnignt downloading (teachers or authors), or other learning activities during downloading time,

3) imported documents show a relative degradation of quality of image and sound; students and teachers may find it acceptable; courseware authors will use them for tests and need a better quality for a final production.

Connecting the DMD with an Integrated Learning System

HYPERGUIDE is an Integrated Learning System (ILS) that provides distance education students with the structure and the objectives of a course, its contents, its document base, and a guidance throughout the learning activities (Bergeron, 1993; Paquette, Bergeron & Bourdeau, 1993). HYPERGUIDE includes integrated communication facilities for students to communicate with their tutor and with other students, and to exchange files over regular phone lines. The DMD has been directly connected to the HYPERGUIDE in such a way that when the student is to make a search, the system asks her if she wants to make a search in the local base or in the distributed one. Further, selected documents from external sources such as DMD can be imported to the database. The integration of Hyperguide and DMD required the implementation of control buttons to invoke DMD, to import selected files and to return to the Hyperguide application. The Apple Events mechanism has been used in programming these functionalities. Moreover, the user interface of the DMD has been adapted to be consistent with the HYPERGUIDE interface, and a management mechanism has been developed to take in charge the documents imported from the DMD in order to integrate them into the resident document base.

Using the DMD in a distance course

HYPERGUIDE is the ILS used in a course entitled "Introduction to Training" at Télé-université. In this course, students are given a classification task where they have to make a search in a document base and compose a structured multimedia document based on the results of their search. These materials (files) include written descriptions, diagrams, spoken explanations, video clips showing particular work situations, etc. Only a small number of files are appropriate in a given situation, and the student's competence is judged from her ability to select the appropriate files. This activity, in order to be meaningful, requires a large file system with a variety of available and reusable materials coming from various sources. Hence the need for the distributed file system described above. In the present form of the course, video documents are analog, and distributed to students through the Télé-université TV channel as well as on videocassettes. For the purpose of this project, the video documents relevant to the classification task have been digitized and made available to the DMD as Quicktime movies.

The classification task is done by students who work at a distance as a team; they access the DMD, import MM documents, exchange files over a telephone network, compose their own MM document, and finally send it to their tutor. A scenario for this teamwork has been elaborated based on the Jigsaw model [Slavin, 1986], in which subtasks of a problem are first assigned to teams of students; when the subtasks are mastered, the teams redistribute so that every new team contains an "expert" for each subtask. Every new team must then find a solution to the posed problem, and finally the solutions are compared and discussed among the teams. This method, proved to be efficient in the teaching of various subjects, was transferred to a Computer Supported Collaborative Learning (CSCL) situation. The resulting scenario, called *Distributed Jigsaw* (DJ), is designed to support distant cooperative learning; it is rich in opportunities for cooperation and reciprocical teaching, with the following six events (Fig. 2):

1. Problem definition and identification of subtasks, normally part of the course instructions.
2. Forming distributed teams and assigning subtasks to the team members by teleconferencing
3. Individual study within teams. Each member becomes an "expert" on a subtask by searching for materials and composing a multimedia document.
4. Teleconference of the "experts" by subtask; here the students compare and elaborate their results together with the members of other teams working on the same subject, leading to modifications in their documents.
5. Return to the original teams; members present their knowledge and documents to the team.
6. Synthesis of results within each team, composition of the solution, i.e. a complex document.

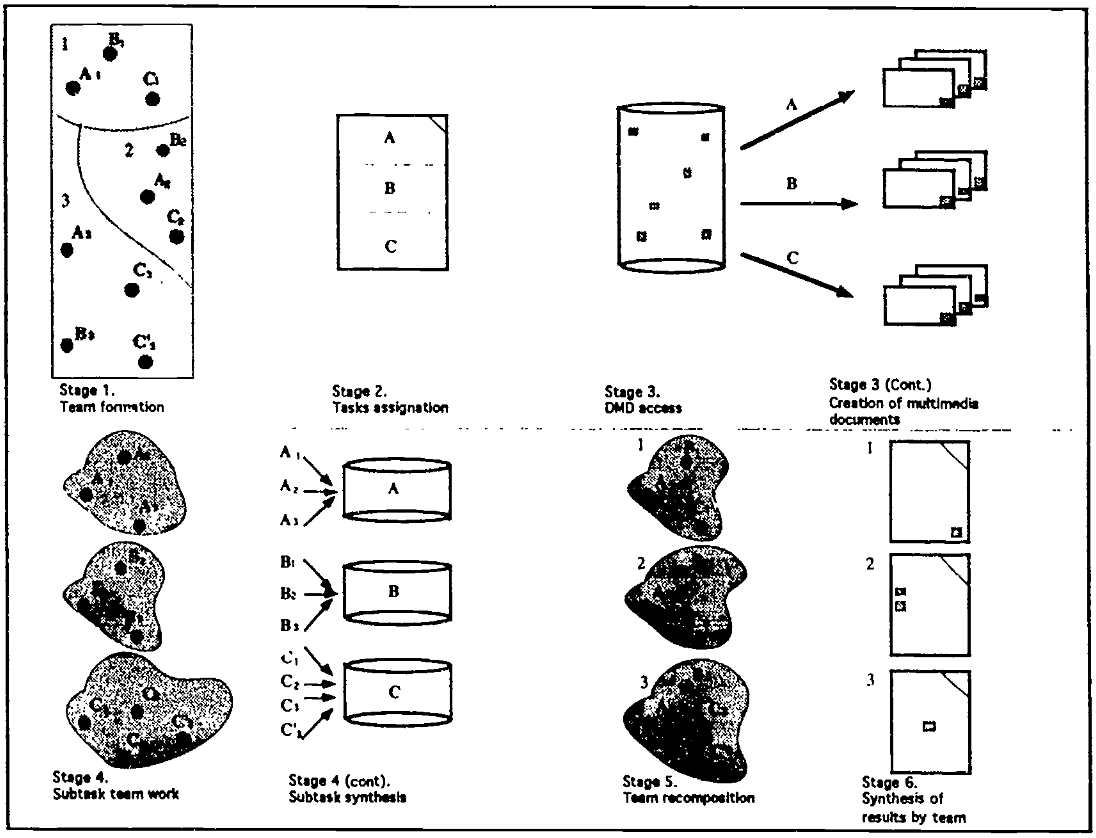


Figure 2. The Distributed Jigsaw scenario

Conclusions and future work

This first experience shows that the availability of high-speed transmission facilities, enabling real-time access to multimedia files, is not essential. The adverse effect of offline access to such files through slow lines can be minimized through proper work organization (e.g. overnight downloading) and through proper use of resources (e.g. freeing the computer during the waiting time). Students, teachers and courseware authors can benefit from accessing MM documents using a low-speed communication infrastructure. The opportunity given to users for negotiating QoS is essential.

A second phase of this project aims at testing the system and experimenting the scenario on a high-speed broadband type of link high speed link between two distant sites, the cities of Montréal and Québec (300 km). The DMD will be implemented under a Windows environment for this purpose. The design of CSCL activities and of the software to support them will be studied in the light of Salomon's reflections (Salomon, 1992).

References

Bergeron, G. (1993). *Hyperguide*, Proceedings of ED-MEDIA'93, Orlando, Florida (poster).

Campbell, A., Coulson G., Garcia, F., Hutchison, D., Leopold, H. (1993). *Integrated Quality of service for Multimedia Communications*, IEEE Infocom'93, San Francisco, March 1993.

Miloucheva, I., and Rebensburg, K. (1992). *XTP service classes and routing strategy for an integated service broadband environment*, Proc. IWACA, Munchen, 1992.

Paquette G., Bergeron G. & Bourdeau J. (1993).*The Virtual Classroom revisited*, Proceedings of Teleteaching'93, August 23-26, Trondheim, Norway.

Slavin, R. (1986). *Student team learning* . National Education Association, Washington, DC.

Salomon, G. (1992). What does the design of effective CSCL require and how do we study its effects? in *SIGCUE Outlook, Special Issue on Computer-Supported Collaborative Learning*. ACM Press, Spring 1992.

Acknowlegdments

We are indebted to Clement Gagnon for technical support, to Stephane Liegeois for connecting DMD and HYPERGUIDE and for digitizing the video documents, and to the Centre de recherche informatique de Montreal for financial support.

Multimedia Training Systems for the Elderly and the Impaired

I. C. BROWN, J. COLLYER, B. CHRISTIE, B. INWOOD, J. KNIGHT, V.P. ASHCROFT AND R. SCANE,
London Guildhall University, 100 Minories, London, EC3N 1JY, UK.

Abstract: The paper describes work recently completed on a European Project which has aimed to use multimedia to help intellectually impaired or elderly people to learn to use Information Technology (IT) at work and in everyday life situations. The objective has been to facilitate the learning of those skills which are needed in order to interact effectively with IT. The target learners are those who, because of their special needs, might otherwise be barred from gaining access to services and potential employment involving the use of IT.

The project described in this paper has created a prototype multimedia learning environment which takes account of the needs of intellectually impaired people and assists potential courseware authors in creating multimedia courses for such users. A model of intellectual impairment has been produced which analyses the needs of the target learner-users. Also a taxonomy of IT situations and a model of access needs are provided which analyse those skills needed to cope with Information Technology in the variety of situations that arise in everyday life and employment. Two prototype pieces of courseware have been produced using the multimedia courseware production environment. The first deals with the task of obtaining cash from an Automated Teller Machine. In the second, the learner-user is taught how to find a book in a library using a computer-based catalogue.

A distinguishing feature of the project is the provision, not simply of a multimedia authoring system, but of a predefined pedagogic model of the system and the provision of support for the author-user in the form of guidance in instructional strategies. In particular 13 "teaching utilities" are identified and these will be discussed in the present paper. In the prototype, these utilities are exemplified as a set of reusable multimedia modules from which a courseware designer is free to select during the creation of a teaching-learning package. The prototype courseware runs under DOS and Windows 3.1 on a 386 or 486 IBM-compatible machine with a touch screen and a video card linked to a laserdisc player.

1. Introduction

It is estimated that, because of improved living conditions and better health care, by the year 2020 there will be between 60 and 80 million citizens in the European Community. One in four of us will be over 60. Disability is associated with increasing age and 70% of the disabled are over 60. Access to Information Technology can help this category of people live independent lives.

The TIDE programme is an initiative of the European Community (EC) to make Information Technology more accessible to disabled and elderly people. Proposals were invited under this scheme from consortia based within the EC with the proviso that at least two European Community countries must be represented. One such contract was won by a consortium involving partners in the UK and the Irish Republic. The project was known as ACT-IT: Application of Computer based systems to Training in IT. The partners were London Guildhall University (formerly City of London Polytechnic) and Interactive Multimedia Systems, Dublin, with sub contracting partners (See acknowledgements).

This paper outlines the work of the ACT-IT project and highlights certain aspects of it. A fundamental aim of the ACT-IT project has been to facilitate access to Information Technology systems by users who by virtue of their special needs, their background or their environment are in some way inhibited from using such systems.

We are all aware of typical reactions to IT devices in which, for example, a 10 year old boy will quickly jump in and feel at home operating a new system (say, a programmable video recorder or home computer) while his 65 year old grandparent has an inertia, perhaps even a fear, which inhibits involvement. Moreover, a sizeable proportion of the population has some special need based on physical or mental impairment which prevents them accessing Information Technology effectively. The physical impairment of being wheel-chair bound may prevent certain users reaching up high enough to access some cash dispensing machines. Intellectual impairment may reduce the capacity of some users to understand and use such devices successfully. There are many people in the lower mental ability ranges who can be helped in their everyday lives and in their potential employment opportunities if they can learn to use IT systems effectively.

2. Support for the Multimedia Courseware Author

In the introduction we have briefly described the target courseware user, but what of the courseware author? Many demands are placed on courseware authors and some of these have been identified by Inwood (1992):

* A detailed knowledge of the domain to be taught
* A knowledge of the group of learners and their needs
* A teaching strategy
* A knowledge of interface design
* A teaching style, and aesthetic and creative abilities.

The ACT-IT project assists the author in all of these areas. It provides resource management aids and gives on-line advice in areas where the author's skills may be deficient. This support is sometimes paper based, but powerful computer based tools and methods are supplied. The advice is based on the model of impairment and of the capabilities of the target user.

In particular a teaching strategy is available for the author to use. A delicate balance is maintained throughout so that the author's creativity is respected and not stifled; on the other hand help and advice is available on demand. Indeed sometimes it will appear on the screen spontaneously.

2.1 The Taxonomy of IT Situations

It may generally be assumed that courseware authors are familiar with the subject area on which they are seeking to create a course. However they may be multimedia specialists and not subject specialists. Moreover they may not be aware of the potential for reusability inherent in individual elements of the courseware they are creating. The ACT-IT system therefore provides a taxonomy of IT-situations and a model of access needs which analyses the skills needed for various IT-related activities. One courseware module may be used in teaching about several IT devices and one environment (eg a library) may require the teaching of many devices.

2.2 The Model of Access Needs and the Model of Impairment

There has been strong psychological theoretic input into the ACT-IT project. The needs of learners seeking to master a wide variety of IT tasks is analysed. This covers reading age and IQ levels required in relation to the overall load on the learner.

Potential trouble spots in teaching situations are identified where excessive loads may be placed on learners because of their particular impairment levels with regard to the following:

* Verbal comprehension
* Memory
* Arousal
* Processing Speed
* Perception
* Attention
* Perceptual Organisation
* Freedom from Distractibility
* Domain specificity

The model of access needs provides a standard load table for any part of a course which indicates, on a six point scale, what demands are placed, in each of the above areas, on a standard learner (e.g. "high"..."low"..."extremely low"). A model of impairment provides, for a particular learner, an impairment profile which will be a six point score with regard to their ability/impairment level for each of the above areas. For each of these areas ACT-IT has devised a set of guidelines to help the author and the delivery system take account of the target learner's potential intellectual strengths and weaknesses.

2.3 Pedagogic Strategies

Examples of pedagogic strategies are available from the authors. Space does not permit inclusion of an example here.

2.4 Design Support Tools

These include the teaching utilities which will be outlined in section 3 of this paper.

2.5 Courseware Production Facilities

The multimedia courseware author is provided with a WIMP interface. A palette of tools is available and authors can manipulate courseware flow templates on screen. Authors may, depending on their response to an initial enquiry about their level of experience, either start with a blank screen, with a standard design in "time line" notation or with a semi-built design (for the moderately competent). In each case authors select tools by icon and position them with the mouse.

Screen templates are supplied, based on good user-interface principles. Authors are free to create their own templates and store them in personal libraries for future use.

2.6 Prototype Courseware

As well as providing production facilities, ACT-IT has provided two demonstration courseware packages. One aims to encourage elderly and impaired bank customers to use cash dispensing machines. The other illustrates how the author can create a course using a set of multimedia building blocks provided. The course is concerned with encouraging library users to use the LIBERTAS computer-based catalogue system to obtain library services. The target user group is the same as described before, namely the impaired and elderly. This prototype illustrates how the author may make selections concerning the use of the teaching utilities discussed in the following section.

3. Teaching Utilities

It is not possible here to give more than a brief introduction to all the work that has been done by the project on the 13 teaching utilities which it has identified. Within the ACT IT deliverables, there is provided, for each teaching utility, a formal definition, a list of guidelines to aid the course author in the use of that teaching utility and the development of media sequences for use with it, and an indication of the kind of control of the utility that the user should be given. The basis for each teaching utility in research and practice is also provided.

In the rest of this section, space only permits a few comments on each teaching utility. A few samples only are provided of references to the literature. For a full discussion, reference should be made to the ACT-IT project final report.

3.1 Attention-grabber

Every good lecture, sermon, TV programme or film must begin with something that grabs the attention of listeners or viewers. Such audiences are always free to walk out or switch off but at least they are in some sense initially 'captive'. In the multimedia context addressed by ACT-IT, the potential user's interest must first be captured: the system may be running in a public place such as a library or bank lobby. Users passing by may not have come with the intention of using the system. So, the purpose of the attention-grabber is to capture attention and encourage potential users to interact with the technology.

Courseware authors should be provided with a library of multimedia attention-grabbers for possible use. Typical examples would include exploding fireworks on screen accompanied by suitable rousing music.

3.2 Personal Tutor

Given the resistance that our target users may have to educational technology, this teaching utility offers them the choice of a human face to guide them throughout the courseware. Such surrogate tutors will talk to the user in a normal human way. They will typically appear on screen at the beginning, giving a welcome. They will help to motivate the learner and introduce the course structure. They may appear later in small windows, or by voice with still images only, to introduce segments of the course, or to provide help where needed.

Ideally the learners should be able to choose their personal tutor from a selection representing different age-groups, races, sexes and cultures.

3.3 Application Context

This provides the learner with motivation for learning a particular skill by showing a situation in which it is desirable to possess it. For example, one might display a long queue at a railway station ticket office with an automatic ticket machine not being used as a train arrives. (British Rail have recently introduced on the spot fines for boarding a train without a ticket, a queue at the booking office being no excuse!)

This teaching utility sets the scene for the target learner and should be made to relate to his or her culture and experience. It is essential that learners find personal meaning and relevance as they experience multimedia courseware. Emotional and motivational appeal is greatly increased when video and sound are used, according to Ambron (1990).

3.4 Peer Group Modelling

The purpose of this utility is to encourage learners to feel more confident by showing members of their peer group succeeding in using an IT-system. These role modelling situations encourage learners to learn by imitating the successes of others and avoiding their mistakes.

As mentioned in 3.3 above, audio-video sequences are recommended but if storage capacity is limited, sequences of still images may be used for this teaching utility.

3.5 Simulation

The simulation is a well established component of technology based education and training. The learner has the benefit of rehearsing actions in an environment that is safe and non-threatening from the point of view of health and safety and also data security.

Users of simulations can acquire skills in domains where the real-life situation would require large amounts of time and or money to set up. Moreover users may learn by making errors in isolation from their real consequences, before using the actual IT device.

3.6 Action Replay

The facility for the learner to replay any courseware sequence should be made available by the multimedia author. This is particularly relevant after using the following teaching utilities: personal tutor, peer group modelling, application context and simulation. Further, the user should be allowed to freeze and unfreeze any part of a sequence as it is replayed.

Repetition is generally regarded as highly important in imparting both declarative and procedural knowledge. Case *et al* . (1986) have emphasised the necessity of recapitulation in remedial teaching. The acquisition of motor skills, for example those required to operate a mouse or Concept Keyboard, makes heavy demands on repetition.

3.7 Notebook

Note taking in lectures has been a traditional component of learning for generations of students. The multimedia environment extends this idea. This teaching utility provides another opportunity for placing the learner in control. The user can define any multimedia item (e.g. a clip of video or a piece of text or music) and store it, with or without personal annotations, in his or her own multimedia notebook. To review the notebook later will obviously be valuable.

Cognitive research into note taking (for example by Suritsky & Hughes (1991) and Bretzing *et al*. (1987) has indicated that taking and reviewing personal notes leads to better retention of material.

3.8 Assessment

Every course needs a method of assessment and multimedia packages are no exception. Computers have been widely used for assessment and so much has been written that the subject can hardly be opened in this short paper. Suffice it to say that the mundane kind of drill and practice activities that characterised early work on CAL should be avoided. Multimedia opportunities should be exploited carefully with due regard to good human factors practice. Sensitivity should be exercised with regard to setting targets for individual users with special needs.

3.9 Reward

An important element of feedback to the learner is some form of reward which acknowledges and reinforces correct responses, effort and improvement.

Use of the Personal Tutor teaching utility is particularly appropriate for delivering rewards. The learner's personally chosen tutor can appear from time to time to give encouragement, either by means of a video clip in a window or by voice only (possibly accompanied by a still image in a window).

Just as the personal tutor is chosen by the learner, so too should be the form of reward. This will depend on the age and ability or impairment level of the learner.

Words of encouragement may be delivered verbally or textually or a points system (preferably animated in some way) should be used. A cartoon figure could replace the human personal tutor. Ideally rewards should be presented in a variety of forms which take account of the learner's impairment; they should never be patronising.

3.10 Analogue Task Map

This teaching utility tells the learner where they are in relation to the course as a whole. It also provides a navigation facility: "Where have I been?" "What have I covered?" "What shall I do next?"

The map could be in the form of a hierarchy diagram of the course, highlighting the units accomplished. In a simple case it might simply be in the form of a thermometer-type linear indicator.

As with all teaching utilities the author has the choice of what forms of the utility to offer the learner or indeed whether to offer the facility at all. In this case, the learner may be provided with the option of opening and closing a map-window at any stage in the course by clicking on a relevant icon.

3.11 Choice

Fundamental to any learner-centred courseware is the concept of user choice. This teaching utility provides the learner with the opportunity to make selections.

Important choices to be offered to the learner include those concerning control over the human-computer interface. Such choices, offered at the beginning of a course would include choice of personal tutor.

Other choices relate to choice of route through the course. Clearly this teaching utility will be used in conjunction with others such as assessment and the analogue task map.

3.12 Tea Break

In industrial relations, it has long been recognised by management as well as trade unions, that greater productivity is achieved if workers are kept happy, comfortable and refreshed by periodic tea-breaks. This teaching utility provides an interlude within a teaching-learning situation so that the learner may enjoy mental refreshment by doing something completely different for a while. Overall performance over a long period will be improved if regular breaks are taken. The frequency and length of the tea-breaks should be carefully controlled by the author.

3.13 Personal Assistant

As with the Personal Tutor, the learner should be presented, at the beginning of the course, with a choice of Personal Assistant. This may take the form of a person (age, sex and race of the user's choice) or a cartoon character or robot. Certain tasks within a course may be too repetitive, tedious, difficult or time-consuming for the learner to carry out. Frustration or anxiety on the learner's part may be alleviated if the author allows delegation to the personal assistant under controlled conditions.

4. Conclusion

It has only been possible in this short paper to outline some of the main features of the ACT-IT project and to spend a little longer on one aspect, namely the teaching utilities. Justice has hardly been done to the project; the written documentation of which, alone, runs to several volumes. It is hoped that further papers and books, based on the ACT-IT approach to multimedia training, will appear soon.

Acknowledgements

The project has been supported by the European Community under the TIDE Programme (Technology Initiative for Disabled and Elderly people) and is known as ACT-IT, TIDE project number 113: Application of Computer-based systems to Training in IT: a multimedia prototype. The project was carried out by two partners: Interactive Multimedia Systems (IMS), Dublin, Republic of Ireland and London Guildhall University, UK. Subcontracting partners included Keith R Philips, Chartered Psychologist and the Electric Brain Company, UK. The work reported in this paper was carried out at London Guildhall University in collaboration with IMS. The University team included, as well as the authors listed above, the following colleagues: Iain Connell, Evelyn Bailey and Heather Smith. The help of all of the above mentioned and of the TIDE Programme of the Commission of the European Communities Directorate-General XIII is gratefully acknowledged.

References

ACT-IT (to be published). *ACT-IT project final report.* Brussels: the TIDE office of the Commission of the European Communities.

Ambron, S. A. (1990). On multimedia in education. In Barker, J. and Tucker, R.N. (eds.) *The Interactive Learning Revolution: Multimedia in Education and Training.* London: Kogan Page.

Bretzing, B.H., Kulhavy, R.W. & Caterino, L.C. (1987). Notetaking by junior high students. *Journal of Educational Research*, 80(6), 359-362.

Case, R., Sandieson, R. & Dennis, S. (1986). Two cognitive-developmental approaches to the design of remedial instruction in cognitive development. *Cognitive Development*, 1 (4), 293-333.

Inwood, B (1992). Supporting the Multimedia Courseware Author: An introduction to the ACT-IT project. A paper presented for and published in the Proceedings of *EdTech '92*, (The Australian Society for Educational Technology), Adelaide, South Australia, October 1992.

Suritsky, S.K. & Hughes, C.A. (1991). Benefits of notetaking: implications for secondary and post-secondary students with learning disabilities. *Learning Disability Quarterly*, 14 (1), 7 18.

Multimedia and Composition: Synthesizing Multimedia Discourse

GARY BROWN
Student Advising & Learning Center
Washington State University
Cleveland 57
Pullman, WA 99164-2105
Browng@wsuvm1.csc.wsu.edu

Abstract: This study examines the multi-modal synthesis of source examples in student compositions. Multimedia is used to present on-screen text and video information on Chinese superstitions and traditions and then to prompt subjects to write. The research compares the number and kinds of examples students produce in their writing. Kind refers to the source of influence, either audio/video or the written transcript of that audio/video. The results demonstrated that the audio/video mode of presentation provides a significantly richer resource for detail than on-screen text for student compositions in the immediate task, but the audio/video resource does not sustain influence in the delayed composition task. These results provide support for the argument that implementation of multimedia in composition instruction merits greater attention to task as well as to the interpretation and analysis of audio/video material if the rich audio/video resource potential multimedia makes available is to be useful in composition instruction.

Introduction and Problem Statement

As multimedia delivery systems for instruction have proliferated, they have found their way into writing intensive courses. The designs as well as the assumptions underlying the goals of integrating multimedia into education vary widely, but almost all suggest, at least in those courses where the outcome of instruction depends upon student writing, that multimedia, and particularly audio/video sources, have influenced that writing. The nature of that influence, however, has not been examined. Meanwhile, popular consensus points repeatedly and suspiciously toward the trend away from the primacy of print. As Maxson contends, "The influences of MTV exert more influence on student text than does the academy" (1993).

Maxson's contention is not unfounded. Ninety-eight percent of American households own a television (Gates, 1993). As Gates observes, "It's MTV more than Montaigne, Viacom more than Voltaire" (Gates, 1993, p. 116). Much television research tends to support both the dominance of audio/video media consumption, and suspicions toward audio/video media's influence upon learning. Contrary to the more positive assumptions underlying multimedia's rapid development and implementation in the classroom, extensive consumption of visual media beyond a threshold of only ten hours per week begins to correlate with weak language skills, a correlation that is pronounced at viewing of over thirty hours per week (Potter, 1987). Graber (1990) has found that we learn very little from watching television news. Meanwhile, our apparent growing reliance upon visual media continues, encouraged by the availability and increasing ease of video manipulations with instantly digitized graphic capabilities now migrating into mainstream word processing programs. The decline of the primacy of the age of print, as McLuhan (1969) forecast, appears to be further accelerated by advanced memory compression and on line-video libraries, not to mention the implications of the rapid development and decreasing cost of hand-held camcorders, video boards, scanners, speech fluid word-processors and even virtual reality. All of these breakthroughs portend at least the transformation if not the demise of composition skills as we have known them.

In spite of the perceived dangers of audio/video as it is represented by television, theories and limited studies claim that multimedia can be used to improve beginning student reading, writing, and general literacy skills beyond the scope of conventional instructional methodologies. The work of DiPardo and DiPardo (1990), Slatin (1990), Bransford et al. (1990) implicitly and overtly counter that media design can be used to encourage and improve the academic performance of students. For instance, Bransford's work argues that there is "more to notice" (1990, p. 124) in visuals than in text. Bransford argues that we can improve the academic performance of those students who tend to rely more on visuals, if, as Graber (1990) also cautions, the visuals are appropriately stimulating and adequately mediated. Still, even the enthusiastic multimedia practitioners acknowledge the difficulty involved in verifying their theories. As John Slatin says, multimedia technologies and writing instruction will require "both a new practice and a new rhetoric, a new body of theory" (1990).

Source Tracking

As multimedia provides an opportunity to mediate visuals, so source synthesis may provide one useful operationalization of multimedia influence upon composition. Research in writing from sources preceding multimedia technologies reflects that need to explore source synthesis, or the sources writers draw upon when they compose. After all, expert writers, research has established, process, select, organize and connect source texts (Hayes et al., 1992; Spivey, 1992). As Spivey notes in her introduction to "Discourse Synthesis: Creating Texts from Texts" (1992), although students who worked from the same texts "produced texts that had some commonalties in content as well as in overall organizational patterns, the writers varied in how they chunked the material, how much material they selected as relevant, and how they connected the material for the reader" (p. 469). Spivey maintains that such "discourse synthesis" is "characterized by its selectivity and its integrativeness" (p. 469). Such selection or choice is particularly important, Spivey maintains, because it must also be integrated to "provide coherence to material from diverse sources and perspectives" (p. 469). This "meaning construction," she argues, "includes both production and comprehension" (p. 470). Though Spivey's work deals exclusively with text, multimedia technologies make the need to explore discourse synthesis more acute.

Concrete Detail

Since multimedia provides an image rich environment, imagery suggests a natural focus for source investigation. After all, the use of images in text has itself held a kind of primacy predating electronic media: "A picture is worth a thousand words." This aphorism has been held as truth even, ironically, in text. As Richard Ohmann observes, citing E. D. Hirsch, Jr. and his "accumulated wisdom of the handbooks," There are only two "maxims" that appear out of a dozen in every major handbook-- "Use definite, specific, concrete language" (1988, p. 353). Ohmann pursues these maxims in detail and into three additional popular handbooks that treat the issue extensively and cite the need to use specific details as a way to "avoid abstract writing" (p. 354) and to make writing "'richer,' more 'vivid,' and more 'intense'" (p. 357). Does multimedia imagery directly influence the transformation of that imagery by students into their own texts? Is what we view as writers as accessible as what we read. The following study was designed to explore multimedia discourse synthesis to begin to assess the influence of multimedia, and specifically audio/video influence upon student writing.

Methodology

The writing assignments examined in this study were designed to reflect a common academic writing task in which students are first asked to read a sample of professional writing and then to write a response to that reading. In this study, however, in addition to exposing subjects to text, full motion laser audio/video was also used as source material for subjects before they were asked to write. The primary dependent measure in both the immediate and the delayed composition tasks was details used in subject's writing and, specifically, the source of that detail as it could be traced to the mode of the message, either the text mode or the audio/video mode presented in the multimedia environment. Secondary dependent measures included recall and time on source.

Forty subjects from Washington State University participated in the study. Four treatments were sequenced to control for message mode and order. Each of the four treatments began with an introductory excerpt from Kingston's "No Name Woman," the subject matter for the second assigned written part of the study. Following the text excerpt from the Kingston source text, each treatment presented one of two possible audio/video supplements on Chinese traditions and superstitions and one of two possible on-screen text transcript supplements on Chinese traditions and superstitions. The text supplements were transcribed from the narrative of the video/narration supplements. The audio/video screens included controls for subjects to pause and review material in order to parallel the recursive nature of reading.

Apparatus, Materials, Stimulus and Facilities

Hardware:

The computer used to present the multimedia treatments was a Macintosh IIci. The laser video player used was a Pioneer 8000 laser disc player. The video and narration were presented using a 19" Mitsubishi color television.

Source Material:

The laser program used as text supplement was the National Geographic™ video, "Portrait of a Hong Kong Family." The program was selected based upon its depiction of themes parallel to those developed in the Kingston essay--superstition and traditions in China and their impact upon sex roles. The program was easily edited and re purposed to support and augment the themes in Kingston's essay/story.

Maxine Hong Kingston's essay/story, "No Name Woman" is excerpted from her novel of the same name in a textbook, Writing About the World.

Multimedia Design:

HyperCard™ and The Voyager Video Stack™ were used to create a linear stack, sequencing the screens subjects saw in each treatment from the first to the last without options. This design was selected because issues of multimedia navigation might have created distracting variables for the purposes of this study, though navigation issues, including providing the subject with control of the mode of delivery, certainly suggest future research. The last of the screens subjects viewed instructed them to write about Chinese tradition and superstition. The prompt encouraged subjects to use concrete examples and to write about a page and half.

Procedures

When subjects finished the multimedia program, they were prompted to describe what they had seen, heard, and read. When they finished, they were presented with the assignment to read the Kingston story/essay, and they scheduled a time to return and write again about the reading. Within a week, each subject returned to write about the influence of superstition and tradition on the role of women China.

Rating subject compositions required two-steps. First, two raters identified details. The cross check reliability was .79. The second step of the rating procedure determined source mode distribution. Two additional source raters maintained a reliability of .95.

In addition, following the composition task subjects completed a multiple choice recall test to control for recall variables. A post hoc written debriefing was conducted to further illuminate the findings.

Results

Immediate Composition Detail Synthesis

Both message order of source mode presentation and recall were eliminated as a significant variables. It was therefore possible to collapse across order to increase the statistical power. (Instead of analyzing ten cases in each treatment, it was possible to statistically examine forty cases by each mode.) After collapsing across order, the design for subsequent analyses was a 2 (mode) x 2 (message) mixed subjects design.

The mean for detail in subjects' writing traceable to the audio/video source was 15.9. The mean for detail traceable to text was 5.1 ($F(1,39) = 51.67$, $p<.0001$). The multimedia audio/video source significantly dominated the text source as an influence in the production of detail in subjects' writing. (See figure 1.)

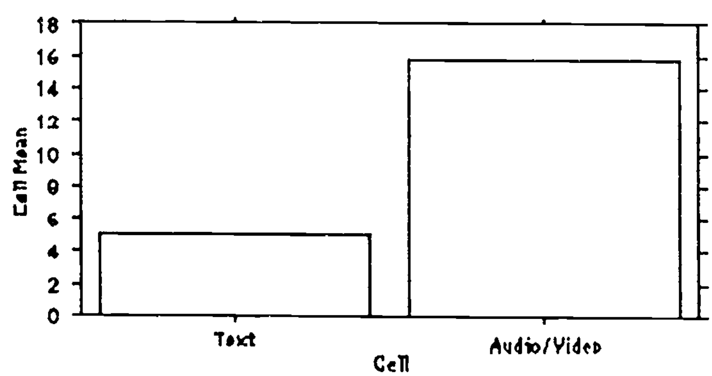


Figure 1. Immediate Composition Source Detail by Mode

Delayed Composition Detail Synthesis

In the delayed composition task, there was no traceable evidence of the multimedia source, either audio/video or text. Since there was zero variance in either the on-screen text or the audio/video variables, a t-test was used to test the main effect of treatment mode to detail production. The effect is significant (t-Value = 10.848, p<.0001). The means for detail traceable to the Kingston source text at 16 was significantly higher than the means for audio/video or on-screen text--both of which were 0. (See Figure 2).

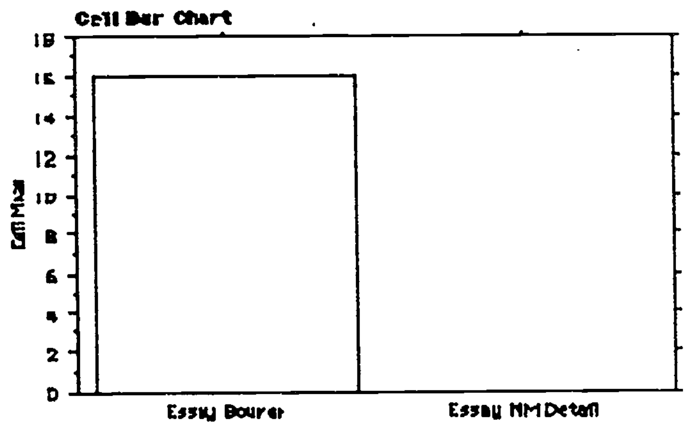


Figure 2: Delayed Composition Source Detail by Mode

Discussion

The dominant influence of the audio/video in the immediate composition task, in spite of the sequence of presentation, supports the contention that multimedia and particularly audio/video "mini-cases" can be rich in information or detail (Spiro and Nix, 1992; Bransford, 1992) and that "the visual material was more 'potent'" than text (Martin & Ditcham, 1987, p. 10). At least during the immediate composition task, students were more than three times as likely to use detail presented on the television screen than they were to use detail presented as text on a computer screen.

Contrary to the audio/video dominance in the immediate composition task, however, the delayed composition task shows no evidence of multimedia influence--audio/video or text. The potential use of audio/video established earlier in this study was supplanted entirely by source text dominance. All of the traceable detail used in the delayed composition task was synthesized from either personal experience (X=6) or the Kingston story, "No Name Woman" (X=16). Since both time on source and recall were controlled for, this finding poses a number of interesting questions for further research.

Perhaps the sequence of the design, a design in which students read the Kingston text *after* experiencing the multimedia treatment, results in recency effects that wash any recall of the multimedia sources. However, the *complete absence* of traceable multimedia influence coupled with the debriefing explanations substantially subordinate or even eliminate recency or other related effects. If the traceable multimedia sources were diminished in the delayed composition task, recency might provide an important explanation. But the few subjects who do refer to the tendency to write about the most recent stimulus tend to qualify their explanations in ways that suggest that it is not an inability to recall multimedia sources that influenced subjects' composition decisions. For instance, in post experimental debriefings, one subject, who described the delayed composition aspect of the experiment as "a typical [freshman English] assignment. . . one that requires time and recollection of facts," said: "It never occurred to me to use the information from the previous day in the essay I wrote today. I tend to focus more on the task at hand and *what's given to me* in regards to the task. It's really strange to me that I didn't even think to use the previous information I received." Several additional debriefing comments corroborated this student's observation.

It may be, of course, that the absence of multimedia source influence simply reflects that the prompt was not specifically clear in suggesting the possibility for using multimedia source material. Grimes explains conflicting findings in his work by pointing to "the nature. . . of the task itself and experimenter instructions. . ." (in press, p. 42). But the assignment did clearly prompt for descriptive detail of Chinese superstition and tradition, stating: "Discuss *in as much detail as you can* . . ." In other words, more than simply a failure of the assignment to sufficiently prompt students to draw upon the multimedia sources, it seems more likely that subjects for some reason do not perceive the "typical" composition assignment as a task that might accommodate material from previously presented multimedia sources.

Perhaps, the contrary findings between the immediate and delayed composition tasks may best be understood by considering the following debriefing response. The subject explains: "I consciously and purposefully drew only upon specific and general examples set up by the reading. Unconsciously, I think the video lent background and insight into my writing. . . . The video gave me a background on the importance of such superstition, but I don't think that it caused me to analyze it any further. In writing the essay, I think I felt obligated or bound to writing about the text."

First of all, notice again that the subject reports memory for the video source. Secondly, though, notice that the subject reports of a perception of the task in which he feels "obligated or bound to writing about the text." One post hoc emerging finding, in other words, was that task representation was instrumental in influencing the kind of information student subjects developed in the delayed composition task. In this case, though, like the subject who felt "obligated" to write exclusively about the text, the perception of the assigned task prohibited the transfer of the previously presented multimedia material--text screen or audio/video. Task, as Flower and Hayes (1988) define it, involves the mental representations that shape our goals, our concept of audience, and the construction of a text itself: "People only solve the problem they give themselves to solve" (p. 93). The "image of one's rhetorical problem . . ." (1988, p. 93) as Flower and Hayes contend, has been significantly influenced by the relationship of previous academic experience, multimedia or otherwise, to the delayed composition task, in this case a task whose representation has been shaped by much more than just the message mode. As one subject reports, "In my writing. . . I want to hit everything relevant to *what I think one wants from my writing*" (Italics mine). Who the subject here regards as audience is not clear, but who or what it is often elicits a sense of obligation that significantly shapes student writing and the synthesis of detail in a way that significantly favors the most recently encountered traditional text.

The findings in the delayed composition study reported here, then, suggest the need to regard task as more than simply an artifact of message mode. Nonetheless, there is also reason to explore the relationship of message mode to the perception of task and not to dismiss it as insignificant. For instance, consider again the subject who said, "The video gave me a background on the importance of such superstition, but I don't think that it *caused me to analyze it any further. . . .*" . Ironically, it seems prudent not to completely disregard the message mode itself, in this case the "image" of the video. Is there, perhaps, an aspect of the video mode that elicits a resistance in subjects to "analyze" the video information "any further," something inherent in the audio/video message mode that reshapes, or fails to shape, "the image of one's rhetorical problem"?

Within the context of this study there is substantial fodder for speculation relative to multimedia implementation and multimedia discourse synthesis. As Spiro et al contend: "A fundamental tenet of all recent theories of comprehension, problem solving, and decision making is that success in such cognitive arenas depends on the activation and appropriate application of relevant preexisting knowledge" (1986, p. 177). Surely it is reasonable to include composition and multimedia discourse synthesis as aspects of "comprehension," "problem solving" and cognitive "decision making." Consequently, it may be wise in future research to consider both the conceptualization of task as well as the conceptual transfer of mode specific "relevant preexisting knowledge" in terms of "appropriate application" constraints.

Study of discourse synthesis writing as Spivey (1992) practices and advocates, and multimedia source synthesis, as Palumbo and Prater (1993) predict, are valuable approaches toward understanding the composition process. The findings in this study may begin to be useful in putting together the inscrutable puzzle of that process and in the design and delivery of multimedia environments. Clearly audio/video information provides a rich source of detail that is available to writers during the composing process. But it appears equally important to understand that there is nothing magical about multimedia sources that will inform the traditional textual analysis without more control of the implementation of the multimedia than this study, a study based largely upon current academic practices, afforded.

Finally, and extending from the previous point, even if the transfer of discreet detailed information from multimedia environments can be harnessed, there is no evidence that quantity of detail alone will improve student writing. In other words, the detail dominance of the audio/video mode apparent in the immediate composition task does not necessary correlate with higher quality writing. In that sense, as critics fear, new media harbors as much threat as promise if it is not re-purposed toward more cognitively complex tasks that include transfer and analysis. The point to point digital nature of audio/video sources, or the "Being There" effects of audio/video sources, provide a great deal of potential for education as well as entertainment, but the pitfalls loom large.

More than simply the mode of the message, it remains the implementation or mediation of the media that seems to really matter. It is a complex task. It will continue to require teachers.

References

Bransford, J. D., Sherwood, R. D., Hasselbring, T. S., Kinzer, C. K., & Williams, S. M. (1990). Anchored instruction: Why we need it and how technology can help. New York: Atheneum.

DiPardo, A., & DiPardo, M. (1990). Towards the metapersonal essay: Exploring the potential of hypertext in the composition class. Computers and Composition, 7(4), 7-22.

Flower, L., & Hayes, J. R. (1988). The cognition of discovery: Defining a Rhetorical problem. In G. Tate and E. P. J. Corbett (Eds.), The Writing Teacher's Sourcebook, (pp. 92-102). New York & Oxford: Oxford University Press.

Gates, H. L. Jr., (1993). The weaning of America. The New Yorker, April 13, 113-117.

Graber, D. A., (1990). Seeing is remembering: How visuals contribute to learning from television news. Journal of Communication, 40(3), 134-155.

Grimes, T. (In Press). Mild auditory-visual dissonance in television news may exceed viewer attentional capacity. Human Communication Research.

Hayes, J. R., Young, R. E., Matchett, M. L., McCaffrey, M., Cochran, C., and Hajduk, T. (Eds.). (1992). Reading Empirical Research Studies: The Rhetoric of Research. New Jersey: Lawrence Erlbaum Associates, Inc.

Hirsch, E. D. Jr., (1977). The Philosophy of Composition. Chicago: University of Chicago Press.

Maxson, J. (1993, March). Multimedia and multivocality in a "remedial" writing classroom. Paper presented at the Conference on College Composition and Communication, San Diego, CA.

Martin, D. S. & Ditcham, L. (1987). Information-processing analysis of television advertisement recall. Journal of General Psychology, 114, 5-12.

McLuhan, M. (1969). A candid conversation with the high priest of popcult and metaphysician of media. In J. L. Golden, G. F. Berquist, & W. E. Coleman (Eds.), The Rhetoric of Western Thought, (pp. 266-296). Dubuque, Iowa: Kendall/Hunt Publishing Company.

Ohmann, R. (1988). Use definite, specific, concrete language. In G. Tate, and E. P. J. Corbett (Eds.), The Writing Teacher's Sourcebook, New York, NY: Oxford University Press.

Potter, J., (1987). Does television viewing hinder academic achievement among adolescents? Human Communication Research, 14(1), 27-46.

Slatin, J. M. (1990). Reading hypertext: Order and coherence in a new medium. College English, 52, 870-883.

Spiro, R. J., Vispoel, W. P., Schmitz, J. G., Samarapungavan, A., and Boerger, A. E. (1986). Knowledge acquisition for application: Cognitive flexibility and transfer in complex content domains. In B. C. Britton & S. Glynn (Eds.), Executive control processes in reading, (pp. 177-199). Hillsdale, New Jersey: Lawrence Erlbaum Associates, Inc.

Sprio, R., & Nix, D (1992). Should computers know what you can do with them? In Nix, D., and Spiro, R (Eds.), Cognition, Education, Multimedia, (pp. 143-163).New Jersey: Lawrence Erlbaum Associates, Inc.

Spivey, N. N. (1992). Discourse synthesis: Creating texts from texts. In Hayes, J. R., Young, R. E., Matchett, M. L., McCaffrey, M., Cochran, C., and Hajduk, T (Eds.), Reading Empirical Research Studies: The Rhetoric of Research, (pp. 469-512). New Jersey: Lawrence Erlbaum Associates, Inc.

Teaching programming to novices: a review of approaches and tools

P. BRUSILOVSKY
International Centre for Scientific and Technical Information
Kuusinen str. 21b, Moscow 125252, Russia. E-Mail: plb@plb.icsti.su

A. KOUCHNIRENKO
Department of Mechanics and Mathematics,
Moscow State University, Moscow, Russia. E-mail: agk@agk.msk.su

P. MILLER
School of Computer Science, Carnegie Mellon University
Pittsburgh, PA, 15213 USA. E-mail: plm@cs.cmu.edu

I. TOMEK
Jodrey School of Computer Science, Acadia University
Wolfville, Nova Scotia, Canada. E-mail: Ivan.Tomek@AcadiaU.ca

Abstract: We review three different approaches to teaching introductory programming: the incremental approach, the sub-language approach, and the mini-language approach. The paper analyzes all three approaches, providing a brief history of each and describing an example of a programming environment supporting this approach. At the end, we summarize the results obtained up to date.

There have been many efforts to develop special methodologies, languages and tools for supporting the initial steps in programming education. A number of special programming environments for novices were designed. In this review we analyze these attempts and summarize the experiences. The review is intended for both the practitioners who are interested in applying effective ways of teaching introductory programming, and researchers who work in the field of programming instruction and learning.

We outline three different approaches to teaching introductory programming. We call them the incremental approach, the sub-language approach, and the mini-language approach. The paper analyzes all three approaches, providing a brief history of each and describing an example of a programming environment supporting this approach. At the end, we summarize the results obtained up to date and discuss directions for further research and development.

The Incremental Approach

The first attempts to overcome the problems of teaching introductory programming use the approach that we call the incremental approach to teaching a programming language. This approach was first time described for a wide audience by Holt et al (1977) for the case of teaching PL/1 and reapplied and extended independently by several authors including Holt's group and others (Holt & Hume 1980; Atwood & Regener 1981; Ballman 1981; Tomek, Muldner & Khan, 1985).

In the incremental approach, the language being taught is presented as a sequence of language subsets. Typically, each subset introduces new programming language constructs while retaining all the constructs of the preceding subsets. Each subset is complete, i.e., it is precisely defined and can be learned or implemented without the following subsets. The whole programming language is then learned by examining progressively larger subsets. At each stage the new constructs and elements of a current subset are introduced to the student and then the new knowledge is mastered by solving a number of programming problems. This permits greater concentration on each set of new language operators and control structures.

Since each subset forms a complete sublanguage, a language processor can be specially constructed to support the learning of each subset. The first idea of Holt et al. (1977) was to take advantage of current subset restrictions to improve error handling and error message generation. Later it was suggested (Ballman, 1981; Tomek, Muldner & Khan, 1985) to combine the advantages of the incremental approach and program visualization. Program visualization is important for novices to more easily understand the semantics of a programming language construct. The problem with this principle is that many aspects of program execution can be visualized, and this requires a lot of screen space and overloads the student's working memory. It also creates implementation problems. Implementing visualization using the incremental approach makes it possible to focus the attention of the student on the new features that are being introduced in the current language subset. This helps to avoid screen overload and student working memory overload and simplifies the design of visualization.

As many language subsets as desired can be specified in order to provide advanced error handling and space for detailed visualization. For example, the incremental approach was suggested for teaching Pascal by Holt and Hume (1980) who divided the language into 8 subsets, Tomek, Muldner and Khan (1985) divided the same language into 10 subsets, and Atwood and Regener (1981) divided the language into 15 subsets. One of the most developed applications of the incremental approach is the PMS system described in the next subsection.

PMS - a system to make learning Pascal easier

PMS (Pascal Made Simple) was developed to help students of computer science understand the semantics of standard Pascal constructs (Tomek, Muldner & Khan, 1985; Tomek & Muldner, 1986). The package was designed in the first half of the 1980's for IBM PC computers and the implementation reflected the limitations of this machine - its small memory and minimal graphics capabilities. The package provided programming environments for ten languages, each of them a subset of Pascal designed to illustrate some aspects of the language in an optimal way. (Hardware limitations made it impossible to make consecutive sublanguages full subsets of preceding subsets.) The individual languages included MINI (primitive data types, I/O, assignment, basic file i/o, control structures), NUMBERS (illustrating real and integer arithmetic and their limitations), TYPES (enumerated types and arrays), RECORDS (records and arrays), VARIANTS (records with variants), SETS, BINARY (binary files), STRINGS, PROCEDURES (procedures and functions), and POINTERS (dynamic data objects and pointers).

Each language has its special display tailored to the aspects of Pascal that it was designed to illustrate, but all shared the same structure to ensure consistency of user interface which was considered very important for novice computer users. In particular, each minilanguage screen consists of a menu line at the top of the screen, a source code window and a data window occupying most of the screen, and an output window on the bottom. Each language features a screen oriented syntax driven editor and an interpreter. The screen editor is capable of intercepting errors as soon as the user types them and starts a new line so as to minimize user frustration with syntax errors and divert their attention from dealing with them and focusing on syntactic details. The editor produces intermediate code executed by the interpreter in either step-by-step or block fashion. The interpreter highlights programming constructs in the source code as they are being executed and the effect is simultaneously shown in the data window. To illustrate the approach taken by PMS to visualize programming concepts, figure 1 shows a program demonstrating the concept of pointers: the data window shows pointers as simulated memory 'addresses' obtained as randomly generated integers.

PMS was used for several years in an introductory Pascal course with about 100 students each year, in weekly 2 hour labs compulsory for all students. The students were asked to execute a series of prewritten programs demonstrating essential concepts, and to write and execute their own programs under supervision. According to both students and instructors, the experience with PMS was very positive and the look 'into the machine' as PMS was executing programs considerably helped students understand the meaning of data types, data structures, pointers, operation of procedures and functions, recursion (PROCEDURES feature a simulated stack), and other concepts. The main reason why PMS

```
PROGRAM Pointers;                                            * p1       <0008>
VAR p1, p2: ^CHAR;                                           * p2       <0009>
          c: CHAR;                                           * c         Q
 BEGIN                                                       *
            WRITE('Enter a character');                      *
            READLN(c);                                       *
            NEW(p1);                                         *
            p1^ := c;                                        *
            NEW(p2);                                         *
            WRITE ('Enter a character');                     *
            READLN(p2^);                                     *
            WRITELN('Characters ',p1^,p2^);                  *
            {Now exchange}                                   ********************
            c := p2^; p2^ := p1; p1^:= c;                    *<0009>  q
            WRITELN('Exchanged: ',p1^,p2^)                   *<0008>  Q
END.                                                         *
*********************************************************************
Enter a character: Q
Enter a character: q
Exchanged: Qq
```

Figure 1: Example of a POINTERS program depicting pointers as simulated memory addresses. Note the I/O window at the bottom showing the only text that the user would normally see on the screen when executing the program in a regular programming environment.

was eventually abandoned was that it did not have enough facilities to support new features of the Borland Pascal taught in the program such as Units and graphics, and there were no resources to update it.

The mini-language approach

The idea of the mini-language approach is to design a small and simple language to support the first steps in learning programming. In most of the existing mini-languages a student learns what programming is by studying how to control an actor, which can be a turtle, a robot or any other active entity, acting in a microworld. Although an actor can be a physical device, the student usually deals with a program model of such a device and observes the behavior of the executive on a screen. A special miniature programming language is used to control the actor. The language includes small set of commands that the actor can perform, and a set of value-returning queries and control structures. Most mini-languages include all basic control structures (conditional execution, looping, recursion, etc.) and a mechanism for creating some kind of sub-program.

The development of the mini-language approach was seriously influenced by *turtle graphics* of Logo (Papert, 1980). Logo was not designed especially for the purpose of teaching programming but the "turtle subset" appears to be a good tool for introducing programming to novices and it provides genuine insight into problem solving with a computer. Note that unlike most of the actors of mini-languages described below, the turtle of Logo is "blind", it can't check its microworld. The "turtle subset" also does not support such classic control structures as Pascal-like *if* and *while*.

The first and still the most popular mini-language *Karel the Robot* was designed by Richard Pattis for university students taking their introductory programming course (Pattis, 1982). Karel contains all important Pascal-like control structures and teaches the basic concepts of the notions of sequential execution, procedural abstraction, conditional execution, and repetition. The overhead of full high level programming languages is reduced as there are no variables, types or expressions in Karel. The actor, robot Karel, performs tasks in a world that consists of intersection streets and avenues, walls, and beepers. Karel can also carry some beepers in his "bag". The main actions of Karel are move, turnleft, pickbeeper, and putbeeper. A set of 18 predicates allows Karel to check the state of his microworld. For example, Karel can determine the presence of nearby walls, if there are any beepers in his bag or at his location, and the direction he is facing. By writing programs that cause Karel to perform carefully selected tasks, students gain experience with the fundamentals while using a pleasant and persuasive metaphor.

Among other mini-languages designed in the beginning of eighties independently of Karel we should mention *Josef the Robot* (Tomek 1982, 1983), *Wayfarer* (Kouchnirenko & Lebedev, 1988), and *Turingal* (Brusilovsky, 1991).

Karel Genie, the mini-language programming environment

Karel Genie is a novice programming environment specifically designed for the Karel mini-language. Karel Genie is a member of the Macintosh based Genie family of structure editor based programming environments developed at Carnegie Mellon University. To support the novice programmer, the Karel Genie provides a set of specially designed tools, which include a structure editor, a program decomposition view for both looking at and editing programs, and a runtime system with advanced visualization tools. The Karel Genie is an integrated programming environment. Editing the program, executing the program and taking a high level view of the call structure are all presented within a single user-interface, allowing students to move from one activity to another with little cognitive overhead.

The details of precise syntax are not a burden for the user of Karel Genie because it is a structure editor. Maximum support is provided to the novice programmer since program construction can be conducted entirely through menu interaction. Every syntactically legal program transformation and only syntactically legal program transformations appear as alternatives in the popular Macintosh pull-down and pop-up menus. As novices develop expertise they begin to exploit Karel Genie's text edit features. The granularity of text edits is kept small and errors tend to be rare and trivial to repair.

Procedural abstraction is especially supported by the Karel Genie graphical program design tool. Students decompose a problem into simpler sub-problems by dragging the mouse from the program root. The environment invites the student to name a new-instruction. (Karel's sub-programs are called new-instructions.) If the new-instruction is not already defined, the Karel Genie builds a new-instruction shell with its name inserted at the correct point. Whether the instruction is new or not, a call to the sub-program is inserted appropriately.

Karel Genie is highly visual and successfully uses multiple representations. This is clearly illustrated in Karel Genie's runtime system. From the point of view of the novice this is not a separate system, but simply an extension of the same environment in which the program was written. Code is highlighted as it is executed. At the moment a "move" is executed in the code window, a Karel icon moves in the world. Selection, repetition, and recursion are similarly visually reinforced. This is extremely valuable in teaching basic concepts as well as in introducing powerful and "advanced" concepts such as loop invariance. The runtime environment also has a visual call stack. This makes procedure invocation and return clear. Coupling this with the fact that Karel programs can be single stepped and run either forward or backward, students who are taught properly in the Karel Genie are using procedures in appropriate ways from as early as he second day of class!

The Karel Genie has been in use in secondary schools and universities throughout the US. for nearly ten years. In addition to CMU it has been used in computer science classes at Harvard University, New York University, Stanford University, Swarthmore College, Ohio State and a number of other institutions.

The sub-language approach

The idea of the sublanguage approach is to design a special *starting subset* of the full language containing several easily visualizable operations. Such a subset can support the first steps of learning programming and helps later in introducing more complex programming concepts. The sub-language approach is quite similar to the mini-language approach, but differs from it in one important feature. While the mini-language approach uses a special miniature language with its own commands and control structures, the sub-language approach provides only a set of commands and queries as an extension of some "big" programming language. These commands and queries are used in combination with standard control structures of the "big" language. The sub-language approach was also influenced by the turtle graphics of Logo. A set of four "turtle" commands of Logo provides the first example of a sub-language. The success of Logo in general and turtle graphics in particular stimulated further research and resulted in several successful attempts to install turtle graphics commands into general-purpose programming languages as PL/1, Pascal Basic, and Smalltalk.

A well less known experience in designing sub-languages has been accumulated in Russia. Russian researchers Zvenigorodsky and Kouchnirenko extended the ideas of Logo turtle and developed a more formal and broader concept of an *executive* which can be applied in both the mini-language approach and the sub-language approach. The executive consists of an actor (the executive itself) working in a microworld and a small set of commands and value returning queries that the actor can perform. The concept of an executive makes possible the use of several executives within the same programming language. One of the first Russian languages for teaching programming - Robik (Zvenigorodsky, 1982) - contains a mechanism for an easy change of executives providing a way to work with different sublanguages within a language. A number of executives can be developed within Robik with their own microworlds, commands and queries. Another approach was developed in the Moscow State University by Kouchnirenko (Kouchnirenko & Lebedev, 1988). This approach and the supporting programming environment is described in the following section.

KuMir - a programming environment with zero response time

Since 1985, learning programming became a part of the high school program in USSR. In the first school textbook, academician Andrew Ershow introduced a simple ALGOL-like language (later nicknamed as E-language, in honor of A. Ershow). The first programming environment for the E-language called E-practicum and it's last version called KuMir are distributed widely in Soviet schools and universities. Several different executives were designed and used in several implementations of E-practicum and KuMir (bileg, landrower, builder, etc). The current school textbook (Kouchnirenko, 1990, 1991, 1993) which has been distributed in some 5.6 millions of copies is based on an improved version of E-language. This textbook introduces the basic concepts of programming by a problem approach. Posing problems to control two executives Robot and Drawer, the textbook naturally introduces such concepts as subprogram, parameter, *loop n times*, *while* and *if* constructs, integer and real variables, arrays, etc.

KuMir is an example of so-called zero response time programming environments. It is an integrated system combining a text editor with a zero-response-time incremental compiler and an unsophisticated source-level debugger. The proper name for such a system is "editor-compiler": while you are typing your program the compiler is working on it, so the program is ready for execution anytime and without any delay. For the first time such kind a system, namely the well-known Cornell program synthesizer, was introduced in (Teitelbaum, 1981). KuMir was developed to reduce the time wasted by teachers and students because of trivial technical problems. The main features of KuMir are as follows.

- All statically detectable errors are found while editing.
- User's and compiler's "domains of influence" are strictly separated. The user may modify source text in the text's window without any restrictions. The only thing the compiler may do is to write messages in the message window. The user may react immediately or postpone the reaction. Thus, the user may transform a program from one correct state to another correct state passing through intermediate incorrect states ignoring all messages.
- Diagnostic messages appear as soon as possible without any special user intervention. Messages appear in all places associated with the error and vanish as soon as the source of the error is eliminated.
- KuMir uses the metaphor of "marginal notes" to attach system messages to the text of the program. The screen is divided into two windows: the left window contains program text, the right window (the "margin") is used for diagnostic messages by the compiler and the debugger. The diagnostic message appears near the line it refers to. While tracing the program, the values of the evaluated expressions are shown in the margins. This visualization feature can be used by beginners to write and debug small programs without input/output commands.

A special feature of KuMir is the possibility of using several executives simultaneously. From a software engineering point of view, the executive in KuMir is a package containing several procedures and functions. External executives can be loaded dynamically. After loading a package, the executive self-integrates into the language: the names of its procedures become visible and KuMir statically checks type correspondence of their parameters. New executives can be designed directly in E-language itself, thus making it possible for the users of the environment to design executives according their own needs. The mechanism of executives substantially extends the range of interesting but solvable problems. Many interesting programs can be divided into two parts: a small header, providing user interface, and a huge package containing tens of procedures doing all the work. If such a package has implemented as an executive, then to design such a program and get some interesting result, the student has to implement only the header itself. For example, to implement a graphical editor over an appropriate executive, the student will usually have to write only 50-80 lines.

Wide experience proved that KuMir, equipped with Robot, Drawer, and other executives is a good tool for the programming courses for 12-16 years old school students. During the last three years KuMir was also used at the Department of Mathematics at Moscow State University to allow students to solve a lot of simple problems quickly: during the first term, the students spend about 20 hours to solve 90 problems on programming, calculus and algebra.

What we can learn from the presented experience

In the above sections, we have described three approaches to teaching programming for novices and presented some experience with these approaches. The three approaches were developed independently and constitute different strategies to make learning programming easier. However, the environments of all three share several similar features stemming from the same goal: to overcome the beginners' problems of learning programming. The analysis of these similarities can provide some good ideas about how to teach programming to novices and how a novice programming environment should be constructed. We distill three lessons from experience:

- Teaching programming should start from a small, simple language subset. Small size and simplicity protect novices from cognitive overload. New features are sensibly added to a well understood base, rather than overwhelming the novice.

- Execution should visually reveal the semantics of language constructs and, where possible, elucidate principles of program structure and execution. Visual queues enable the novice to understand semantics of introduced constructs, protecting them from developing misconceptions. Visibility provides a feedback for exploratory learning and problem solving.

- Concepts need to be embedded in rich contexts. Visual metaphors make it easier to develop interesting problems for important concepts. Problems that achieve visible and meaningful results aid concept mastery by reinforcing with problem solving activities. Newly learned concepts should be immediately applied to solving meaningful attractive problems.

Another important lesson is that selecting the best approach is not enough for successful teaching of programming. Even with the right approach, learning programming should be supported by a good programming environment. Such an environment should keep both the microworlds and the student program visible on the screen. The program should typically be executed one instruction at a time, while the interpreter highlights programming constructs in the source code as they are being executed and the effect is simultaneously shown in the microworld. For the mini-language and visible subset approaches, the interpreter should also provide visualisation for those concepts of the language that are not visualized by the microworld. Important features include visualization of variables and the stack of subroutine calls. Such an interpreter supports both understanding of programming constructs and program debugging. The environment should also provide a structure editor, to increase student productivity, and enable the student to concentrate on the more important parts of problem-solving. The structure editor protects the student from making most syntax errors and provides immediate diagnostics for the remaining ones. Providing menus or hot keys, the structure editor also solves the contradiction between the requirements to enter the constructs easily and giving them meaningful names.

Two other important points to discuss are the areas of applicability of the three approaches and main differences between them. The main distinction between the three approaches is whether the approach uses a special executive or special visualization as in the mini-language and sub-language approaches, or regular language operations visualized by the environment as in later implementations of incremental approach. The advantages of the first two approaches are in attractiveness and additional motivation which are provided by a well-designed visible executive. Programming problems applied to the actor in a microworld are usually more attractive and meaningful for the students. Sometimes these problems look as a puzzle rather than a 'serious' programming task and the problem-solving activity becomes a kind of a game. We think that if the novices being taught are well-motivated computer science students then an executive or a special visible subset are not vitally required, but even here they can help students in their first programming lessons. For younger students and for those who learn programming as a part of some computer literacy course or on their own, the application of a motivating executive is very important for the success of learning. The younger the students are the more attractive the executive should be.

The second distinguishing point is whether an executive or a visualization subset is used as a part of a bigger language such as Logo or Pascal as in the sub-language approach, or whether a special miniature language is designed to control the executive. We think that the sublanguage approach is better when the student's direct goal is to learn a particular big language. The student can learn most of the control structures and operators of the language more easily with the help of the visible subset. However, if the goal in not to learn a particular language but to learn the principles of problem solving by programming which can later be followed by learning a "real" language, the mini-language approach is better. Mini-languages can provide a sound basis for systematic problem solving for people who will program only to customize their spreadsheet, database, or CAD package, or another application program. Mini-languages open a door to new educational opportunities. Regardless of the student's eventual penetration into programming and regardless of the age of the student there is a positive residue of the study of a mini-language.

Other paradigms

In conclusion, we should mention that this paper discusses only the approaches related with teaching procedural paradigm languages. The lessons learned are, however, important for teaching other programming paradigms as well. Let us see, for example, what has been done in supporting object oriented programming which is becoming more and more popular. The most famous example of an object-oriented programming language is probably Smalltalk (Goldberg & Kay, 1977), for which one of the main motivations was to transport programming closer to the human perspective of the world. However, even though Smalltalk was often advertised as a programming language very suitable for teaching programming, it appears quite difficult for novices (Singley & Carol, 1990). Special object-oriented mini-languages (Fenton & Beck, 1989; Sellman, 1992) as well as an analog of a "visible extension" of Smalltalk (Borne, 1991) were suggested to provide an "easy introduction" to object-oriented programming. The concept of a special environment with extended visualization is also important for the object-oriented approach. A good example of how such an environment can be designed to support the first steps of learning Smalltalk is presented in (Böcker & Herczeg, 1990). Another language and environment that should be mentioned in this context is Prograph (Cox, 1989). Prograph - standing for programming in graphics - is a general purpose object oriented programming language currently running on Macintosh computers. Prograph's interface is purely graphical and supports visual execution. The language has been used to teach high school students.

References

Atwood,J.W. & Regener,E. (1981). Teaching subsets of Pascal. *SIGCSE bulletin*, 13(1), 96-103.

Ballman,T. (1981). Computer assisted teaching of FORTRAN. *Computers and Education*, 5(2), 111-123.

Böcker,H.-D. & Herczeg,J. (1990). Browsing through program execution. In D.Diaper et al (eds.) *Human-Computer Interaction - INTERACT'90*. Amsterdam: North-Holland, 991-996.

Borne,I. (1991). Object-oriented programming in the primary classroom, *Computers and Education*, 16(1), 93-98.

Brusilovsky,P. (1991). Turingal - the language for teaching the principles of programming. In *Proceedings of the Third European Logo Conference*, Parma: A.S.I., 423-432.

Cox,P.T., Giles,F.R. & Pietrzykowski,T. (1989), Prograph: a step towards liberating programming from textual conditioning. In *Proceedings of 1989 IEEE workshop on visual languages*, 150-156.

Fenton,J. & Beck,K. (1989). Playground: An object-oriented simulation system with agent rules for children of all ages. In Object-oriented programming: systems, languages, applications. In *Proceedings of OOPSLA '89*, 123-137.

Goldberg,A. & Kay,A.C. (1977). Teaching Smalltalk: Methods for teaching the programming language Smalltalk; Smalltalk in the classroom, Xerox PARC, June 1977.

Holt,R.C. at al. (1977). SP/k: A system for teaching computer programming. *Communications of the ACM*, 20(5), 301-309.

Holt,R.C. & Hume,J.N.P. (1980). *Programming standard Pascal.* Reston.

Kouchnirenko,A., & Lebedev,G. (1988). *Programming for mathematicians.* Moscow: Nauka, (In Russian).

Kouchnirenko,A., Lebedev,G. & Sworen' R. (1990 ,1991, 1993) *Foundations of Computer Science and Technology.* Moscow: Prosveshchenie. (In Russian)

Miller,P. & Chandhok,R. (1989). The Design and Implementation of the Pascal Genie, In *Proceedings of the ACM Computer Science Conference,* Louisville, KY.

Papert,S. (1980). *Mindstorm, Children, Computers and Powerful Ideas.* New York: Basic Books.

Pattis,R.E. (1981). *Karel - the robot, a gentle introduction to the art of programming with Pascal.* New York: Wiley.

Sellman,R. (1993). Gravitas: An object-oriented discovery learning environment for Newtonian gravitation. In *Proceedings of East-West conference on human-computer interaction,* Moscow, 31-34.

Singley,M.K. & Carrol,J.M. (1990). Minimalist planning tools in an instructional system for Smalltalk programming. In D.Diaper et al (eds.) *Human-Computer Interaction - INTERACT'90.* Amsterdam: North-Holland, 937-944.

Teitelbaum,T. & Reps,T. (1981). The Cornell Program Synthesizer: A syntax-directed programming environment. *Communications of the ACM,* 24(9).

Tomek,I., Muldner,T. & Khan,Ş. (1985). PMS - a program to make learning Pascal easier. *Computers and Education,* 9(4), 205-211.

Tomek,I. & Muldner,T. (1986). *A Pascal primer with PMS.* McGraw-Hill.

Tomek,I. (1982). Josef, the robot. *Computers and Education,* 6, 287-293.

Tõmek,I. (1983). *The first book of Josef.* Englewood Cliffs: Prentice-Hall.

Zvenigorodsky,G. (1982). Introductory language Robic in an educational programming environment. In *Software for informatics.* Novosibirsk, 72-85. (In Russian)

123

R-WISE: A Computerized Environment for Tutoring Critical Literacy

P. CARLSON
U. S. Air Force Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235, USA
E-Mail: Carlson@rnd.brooks.af.mil

M. CREVOISIER
Command Technologies, Inc.
U. S. Air Force Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235, USA
E-Mail: Crevoisier@rnd.brooks.af.mil

Abstract: This paper describes a computerized environment for teaching the conceptual patterns of critical literacy. While the full implementation of the software covers both reading and writing, this paper covers only the writing aspects of R-WISE (Reading and Writing in a Supportive Environment). The project is part of a seven-year Air Force effort -- the Fundamental Skills Training project -- to transition advanced computer-aided instruction to the public school sector. This paper gives an overview of the approach the tutor uses, its underpinnings in cognitive and textual theories, and the results of a pilot study completed in 1993.

Background and Purpose

The national attention focused on the "literacy crisis" addresses a significant and real problem in contemporary America -- the alarming increase in numbers of people who simply cannot read. However, deficiencies in composition are just as noteworthy. Most adolescent learners write only on a minimal level of acceptability. Furthermore, their inadequacy is not necessarily a result of poor spelling, vocabulary, grammar, verbal fluency, syntax, paragraphing, or other such production skills. Bereiter and Scardamalia call the form of writing practiced by inexperienced writers *knowledge telling* (1987). For these two noted researchers, *knowledge telling* is characterized by (1) a simple task execution involving only limited planning and minimal mental engagement, (2) production methods adapted from oral abilities, (3) organizational patterns based on free-association or simple narration, (4) development that contains large chunks of irrelevant information or elaborations based on simple descriptions, and (5) an egocentric perspective. The antithesis, *knowledge transformation*, as practiced by good writers, is characterized by (1) guided planning and situational diagnostics, (2) rich mental representations of text possibilities for a wide range of scenarios, and (3) a robust "executive control program" for allocating mental resources and for handling the tremendous cognitive load of verbal composition.

Computers and the Teaching of Writing

The notion of using computers to facilitate the teaching of writing has been around for quite some time (Bangert-Drowns, 1993). In general, one can see two broad trends in the early efforts: (1) research into whether or not word processing alters composing, and (2) various forms of parsers and natural language processing for detecting flaws in text (such as the many separate programs in the Writer's Workbench®). Both efforts have produced only limited results because both technologies -- at their roots -- are intended to enhance productivity rather than facilitate cognition. In short, they are just not the type of tool -- as delineated by Gavriel Salomon (1993) -- to produce a lasting gain in performance, *once the tool has been removed.*

R-WISE encourages students to practice composition in a computer-mediated environment, where the specially-designed software acts as a *procedural facilitator*. This term is used by Vygotsky (1978) to explain the cognitive mentoring and developmental dynamics that occur between master and apprentice and between peers during collaboration. Salomon (1988) and Zellermayer (1991), among others, uses the term to suggest that the computer can serve as a peripheral brain for the fledgling student and provide the scaffolding that allows the novice to practice the more robust problem-solving behaviors of an expert. R-WISE serves as a *cognition facilitator* for critical thinking by:

- Easing demands on short-term memory and helping to focus attention on strategically important aspects of writing;
- Guiding the inculcation and self-initiation of higher-order processes (metacognition) which the novice writer is unlikely to activate without prompting;
- Explicitly modeling strategic intellectual processes so that the fledgling student avoids what Collins & Gentner (1980) have termed "downsliding," or becoming increasingly entangled in lower and lower levels of mental actions, finally concentrating all mental energy on such things as spelling, grammar, and sentence construction to the exclusion of larger concerns in the process.

Software Components

R-WISE consists of a suite of computerized "tools" to aid ninth-graders in learning the art of prose composition. We selected the tools based on (1) their potential to represent components of the writing process as a visual algorithm and (2) their ability to model robust expert behaviors for the student. A simple model for the composing process that has gained wide acceptability in pedagogy depicts writing as having three central phases or stages: (1) Prewriting or invention, (2) Writing or drafting, and (3) Revision or editing. The three components showcased in this paper mirror this partitioning of the process.

- Cubing (Prewriting) -- Information is not knowledge, just as understanding the content of a piece of prose is not equivalent to understanding and using the concepts presented in a text. This tool -- based on a graphical representation of complex mental operations -- helps the student to draw inferences and to elaborate on the basic concepts being developed in a piece of emergent text.
- Idea Board (Drafting) -- This tool mediates a major cognitive shift in the writing process -- from the macro structures of thought to the micro structures of socially accepted, connected prose. Using a visual algorithm, the tool models robust, expert behaviors for writing a first draft. By foregrounding activities at appropriate times and relegating others to less prominence, the tool teaches the student to manage the cognitive load of composition.
- Revision -- This tool helps the student to "re-see" a completed draft. Editing, as fostered in this tool, refers to substantial changes, such as improving style, adding to or subtracting from the content, rearranging parts, or completely re-writing. These more global, deep-structured editing acts are associated with higher-order cognitive skills (e.g., discerning patterns in bodies of information, exercising judgment, analysis, synthesis).

Instructional Approach

R-WISE uses a hybrid paradigm for interactive instruction. Part of the guidance comes from adaptive tutoring using traditional AI formalisms and part of the teaching comes from the powers of reification (or representing complex processes as manipulable objects on the computer screen). Tools (1) accommodate deficiencies and thereby reduce frustration for a weak writer, (2) emulate some of the crucial functionality of paper copy, (3) enrich the environment and thereby sustain motivation, and (4) model robust behaviors. While each of the three tutors being considered concentrates on a specific cluster of skills, all three have a unified method for delivering this layered instruction and a canonical architecture for the software and the interfaces. Figure 1 guides the discussion for the next six subheadings.

Setting Goals: (Area 1 of Figure 1) Using a *knowledge-telling* approach, the novice writer views composition as if it were a straight-forward exercise in generating ideas through association. For the expert, however, having an explicit, stable set of goals fosters a kind of filtering activity that focuses the mental task from the outset.

Each of the three cognitive tools handles this concretizing of goals in a slightly different manner. In *Cubing*, the tool selects from a collection of 9 possible readers and 9 possible reasons to write, (e.g., write a persuasive piece for a reader who is an expert in the subject matter). The student is then given this combination as a rhetorical situation. *Idea Board* asks the student to pick from a list the most applicable response to each of four questions that identify the purpose, the audience, the text form, and anticipated difficulties. *Revision* asks the student to set "sliders" to indicate three dimensions of the reader and three dimensions of the writer.

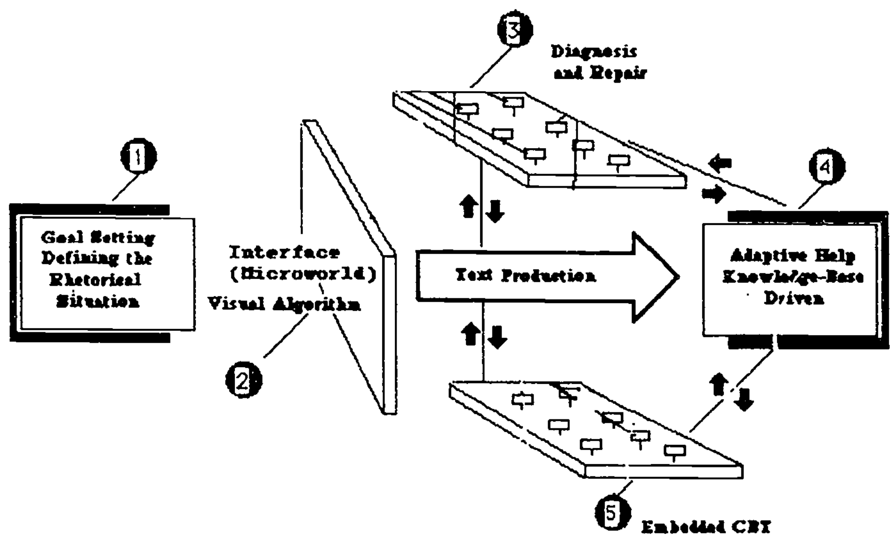


Figure 1: System overview of hybrid tutoring capabilities

Not only does this exercise help the novice student focus on an area where she is weak, this preliminary work "sets" the parameters of the adaptive tutor. Each writing environment now has a "frame" or backplane of conditions against which further actions can be evaluated during the remainder of the session on the tool. (If the student changes goals, the frame is also updated.) Table A gives the number of combinations (or rhetorical situations) tracked by each of the tools. Clearly, the repertoire is rich, and becomes even richer as these preliminary combinations are conjoined with additional data points drawn from the student's subsequent activities as the writing progresses.

Table A

Tracking Combinations in the Frame

TOOL	CONDITIONS	COMBINATIONS
Cubing	Reader Profiles (9) Aims or Purposes (9)	81
Idea Board	Purpose (5) Audience (6) Form (6) Difficulties (7)	1,260
Revision	Reader's Age (3) Reader's Knowledge (3) Reader's Attitude (3) Writer's Distance (3) Writer's Tone (3) Writer's Purpose (3)	729

Visual Algorithms: (Area 2 of Figure 1) The second way in which R-WISE teaches is to reify complex and potentially covert mental operations. Where possible, the interfaces of R-WISE represent visual organizers for specific intellectual processes. As explained by J. H. Clarke (1991, pp. 526-7), ". . . [f]rom the standpoint of cognitive theory, graphic frames mimic aspects of semantic memory structures or schemata, that learning theorists believe organize the mind." For example, "Idea Map," an early workspace in Idea Board that encourages structured brainstorming by presenting the student with a visualization of mental manipulations. Given the premise that most of the clients for R-WISE probably have learning preferences that are concrete/visual rather than abstract/language, we provide "objects" for obscure mental actions. Similar to "webbing" -- a paper-and-pencil technique used in the writing classroom -- this thinking frame prods the student to cluster ideas into proto-paragraphs. Working with the "Idea Map" helps the student to recognize and to take control of the intellectual processes foundational to composing.

Diagnosis and Repair: (Area 3 of Figure 1) Writing is a complex, multi-dimensional activity analogous to a contingency management problem. Only in working through candidate solutions does the nature of the problem become fixed, or even definable. Rather than working in a linear fashion through the model of PLAN - PREWRITE - DRAFT - EDIT, good writers use an opportunistic approach. They constantly measure the emerging text against a set of expectations, while at the same time recognizing and capitalizing on serendipitous gains, weaving these "discovered" possibilities into a new rendition of the overall plan and product.

Unfortunately, text production strategies for novice writers are frail and one-dimensional; such impoverished capabilities do not lend themselves to interruptions or re-assessments. As evidenced by Bereiter and Scardamalia's research into the writing process for novices, little evidence can be found that weak writers can participate in self-cueing or self-monitoring activities while engaged in a production of text (1987). In fact, the very act of breaking out of their one-dimensional, stream-of-consciousness mode jeopardizes the continued production of text. Diagnosis-and-Repair is an evaluation loop that partners with the student to reduce the cognitive load and that encourages the student to enter into an assessment episode. This loop -- essentially modeled after Bereiter and Scardamalia's well-researched CDO (COMPARE, DIAGNOSE, OPERATE) sequence (1987) -- takes a very sophisticated, open-ended problem and pares it down to a manageable set of options for the inexperienced writer.

Clearly, the lists of options are a form of embedded instruction, and the student probably will come away from extended exposure to any of the three tutors with better content knowledge about what can go wrong at various stages of composition. However, we feel that the more important lesson the student learns is an enriched self-regulatory capacity so that she can move out of the text production mode into a higher-level cognitive activity without disrupting the whole composing process. This ability to suspend operations on one level and to focus mental energies on another is characteristic of the experienced writer (Hayes and Flower, 1980).

Adaptive Advice: (Area 4 in Figure 1) In the metacognitive stage (diagnosis and repair), the machine partners with the student to develop the sensitivity and awareness necessary to know what is wrong with a prose performance and how to improve the result. Yet, because the diagnostic is performed by the student, there is a potential for a mis-judgment. Additionally, if the system is to serve as an intelligent "guide" or "coach," the tool should have a feedback loop to indicate the "reasonableness" of the course of action the student is pursuing, baselined against some known set of criteria.

Adaptive advice adjusts its statements based on an "intelligent" assessment of the situation -- meaning that the software compares the manipulation the student is working on with the conditions of the frame and determines how "correct" these actions are given the circumstances. The resultant prompts help the student to learn the more subtle aspects of adaptation to audience and purpose. They also help the student to stay on the right track and avoid the frustration of writing text that is later deemed to be inadequate to the task. *Cubing*, *Idea Board*, and *Revision* contain advice that is germane to the focus of the tool. For example, *Cubing* -- whose domain is prewriting or the invention phase of writing -- prods the student to generate ideas. *Revision*, on the other hand, contains advice to aid in the assessment of such things as coherence, introductory and concluding paragraphs, whole-paper arrangement, paragraph structures, and effectiveness of individual sentences.

For all the intelligent tools, adaptive help is generated through a kind of triangulation, based on the rhetorical situation (frame) and the moves made by the student in the microworld or visual workspace. Monitoring the combination of rhetorical situation and place in the writing process creates a rich representation for accessing instructional statements. Table B shows the number of instructional situations captured in each tool. Because of the large numbers, in the current version of R-WISE, we have implemented a pruning algorithm that weighs the various elements going into the instructional situation and generates a manageable set of instructional statements. For example, the student (having written a topic sentence for a paragraph of factual detail intended for an audience at a distance from the writer, younger than the writer, and less knowledgeable than the writer) might be prompted to consider whether or not the advanced organizer is adequate for the purpose and the audience.

Table B
Generating Adaptive Help

TOOL	FRAME COMBINATIONS	DIAGNOSTIC CHOICES	INSTRUCTIONAL SITUATIONS	ADVICE STATEMENTS
Cubing	81	18	1,458	72
Idea Board	1260	18	22,680	50
Revision	729	27	19,683	153

Just in Time Tutoring: (Area 5 in Figure 1) While designing R-WISE, we carefully planned how to integrate the tutor into a year-long ninth-grade curriculum. As currently fielded, the tutor takes up about 20-25% of the course. The production skills necessary for writing (e.g., topic sentences, paragraph patterns, conclusions and introductions, and other rhetoric fundamentals) are taught in the classroom, not on the computer. This is a deliberate decision. To act as an accelerator, the computer has to support the *process* of literacy. Interrupting the process to teach the enabling skills (1) mixes levels, styles, and purposes of instruction, (2) creates breaks in the train of thought from which the student cannot recover, and (3) results in a fairly unexciting electronic workbook.

While production skills and metacognitive skills are not interchangeable, they are correlated in that they must occur simultaneously in expert behaviors. After diagnosing a problem and getting a repair, the student may still be at a loss as to what to do. Recognizing that students may need reminders of materials covered in class, we have embedded a CBT component in R-WISE which provides "hints" upon request. This instruction (similar to a high-end form of context-sensitive help) is analogous to a *job aid* in that it gives a synoptic overview of concepts presented in class. Its purpose is to serve as a reminder or a refocusing prompt for the student rather than a full-blown instructional component. All three tools have a just-in-time-tutoring (JITT) module for each diagnostic choice they present. Each unit is terse and highly visual, representing fundamentals of composition with conceptual maps and pseudo-animations.

Pilot Study and Future Plans

A pilot version of R-WISE was field-tested during academic year 1992-1993. A San Antonio, Texas, high school with a population of approximately 650 ninth-graders used various components of the system over a nine-month period. In working with this Beta-test site, we were interested in a variety of research questions, including issues of design and user acceptance. We used several instruments to measure learning outcome, but only the results of the writing sample are presented here.

A second high school, within the same district and demographically similar to our pilot site, was selected as a control. Two equivalent writing prompts were devised, and one was given in January as a pre-test; the other was given in May as a post-test. We devised a rubric for holistically scoring the papers on a 1 to 6 scale and then had the approximately 2,200 samples professionally evaluated. The standard procedure of having two readers examine each paper was used. Inter-rater reliability was .79. The differences in means on pre-test and post-test for both control and treatment were deemed statistically significant using t-tests and are given in Figure 2. We are encouraged by the approximately 7% gain accomplished by the treatment group, but are cautious in our interpretations given that the software was in a beta version during the pilot study. A more finished version of R-WISE is now being used by nine test sites in five different states. We hope to get even more of an effect from this production version of the software.

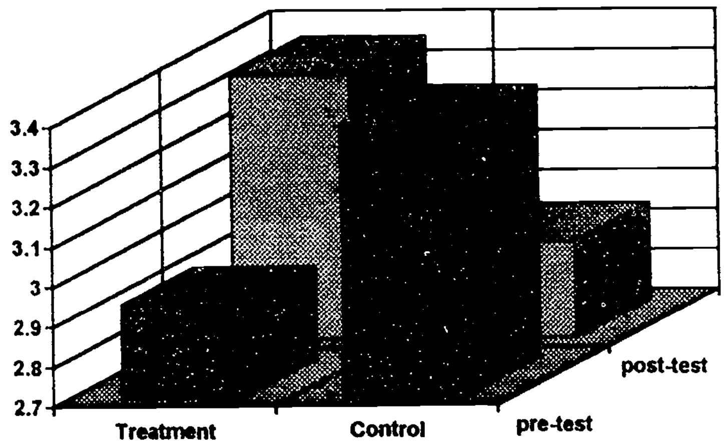


Figure 2: Mean scores for control and treatment on pre and post tests

References

Bangert-Drowns, R. L. (1993). The word processor as an instructional tool: A meta-analysis of word processing in writing instruction. *Review of Educational Research*, 63(1), 69-93.

Bereiter, C. & Scardamalia, M. (1987). *The Psychology of Written Composition.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Clarke, J. H. (1991). Using visual organizers to focus on thinking. *Journal of Reading*, 34(7), 526-534.

Collins, A. M. & Gentner, D. (1980). A framework for a cognitive theory of writing. In L. W. Gregg and E. Steinberg (Eds.), *Cognitive Processes in Writing* (pp. 51-72). Hillsdale, NJ: Lawrence Erlbaum Associates.

Hayes, J.R. & Flower, L.S. (1980). Identifying the organization of writing processes. In L. W. Gregg and E. Steinberg (Eds.), *Cognitive Processes in Writing* (pp. 3-30). Hillsdale, NJ: Lawrence Erlbaum Associates.

Salomon, G. (1993). On the nature of the pedagogical tool: The case of the writing partner. In S. P. Lajoie and S.J. Derry (Eds.), *Computers as Cognitive Tools* (pp. 179-196). Hillsdale, NJ: Lawrence Erlbaum Associates.

Salomon, G., et al. (1991). Partners in cognition: Extending human intelligence with intelligent technologies. *Educational Researcher*, 20(3), 2-9.

Salomon, G. (1988). AI in reverse: Computer tools that turn cognitive. *Journal of Educational Computing Research*, 4(2), 123-139.

Vygotsky, L. (1978). *Mind in Society: The Development of Higher Psychological Processes.* Cambridge, MA: Harvard University Press.

Zellermayer, M., Salomon, G., Globerson, T., & Givon, H. (1991). Enhancing writing-related metacognitions through a computerized writing partner. *American Educational Research Journal*, 28(2), 373-391.

Acknowledgments: This paper does not necessarily reflect the opinions or policies of the U.S. Air Force. Elaine Hitzfelder, Tim Hudson, D'Ann Johnson, and D'Anne Redmon -- all teachers at MacArthur High School in San Antonio, Texas -- served as subject matter experts during the development of R-WISE.

The Development of Computer Assisted Learning in UK Universities

JOHN CASTLEFORD AND GEOFF ROBINSON
Computers in Teaching Initiative Centre for Geography
Department of Geography, University of Leicester
Leicester, LE1 7RH
United Kingdom
Email: cti@le.ac.uk

Abstract: The paper reviews two of the UK's national programmes to promote the use of computer-based teaching throughout higher education. In the first phase (1985-9) of the Computers in Teaching Initiative (CTI), a suite of 139 individual software-production projects generally failed to meet expectations. In 1989 CTI became a network of subject-based centres with a remit to promote the use of information technology (IT) within specific academic disciplines. Following wholesale changes in the national organisation of higher education in 1991, the UK government sought to double the number of students in higher education. A new programme of courseware development was initiated, the Teaching and Learning Technology Programme (TLTP): 75 projects are currently being funded, involving both single academic institutions and consortia. In examining the effects of these initiatives it is concluded that, while computer-based learning (CBL) has pedagogical benefits, there are many factors which hinder its adoption by university teachers.

The development of innovative teaching and learning materials in higher education within the UK invariably assumes an increased role for information technology (IT). The Computers in Teaching Initiative (CTI) and the Teaching and Learning Technology Programme (TLTP) have national remits to promote the use of IT in university teaching. This paper reviews the progress of the three distinct approaches that have been followed:

(1) CTI Phase I: largely concerned with software production;
(2) CTI Phase II: subject-based centres within academic disciplines; and
(3) TLTP

(1) The Computers in Teaching Initiative. Phase one: 1985-1989

It was generally accepted throughout the 1970s that computers were being under-used in teaching in UK universities. As a result central funds were earmarked to provide more computing resources for use by undergraduates, typically hardware in the form of multi-access minicomputers. In the bidding process which followed science and engineering were particularly favoured. However, the rapid introduction of microcomputers in the 1980s changed the nature of academic requirements somewhat, as did the realisation that very little effort was being focused on utilising the general educational potential of computers for teaching. Although computer-based learning (CBL) was beginning to emerge, very little suitable software was developed.

Overall responsibility for computers within higher education in the UK was vested in the Computer Board for Universities and Research Councils (CBURC) which immediately saw the need to develop not only software but also support materials for teaching purposes. Through a new initiative, CTI, funds were made available to develop and produce the resources needed throughout higher education. CTI's main aim was to promote computer-based learning and training. This would involve (a) assessing the pedagogical potential of IT; (b) promoting increased awareness of the potential benefits of IT for university teachers, students and administrators; (c) assessing needs throughout higher education; and (d) producing and implementing educational software. Clearly the focus had shifted from hardware to software. Organisational and logistical issues were also considered important issues, as were techniques and standards for evaluating software. Funds were disbursed through a competitive bidding process. Although the overt objective was to provide for more software development, the underlying ethos was concerned with influencing the ways in which students were taught and engendering a more receptive environment for CBL. By 1987 some 139 projects had been approved from a total of 700 bids received and at least one project was awarded in each university (Table 1). Most projects were centred within individual university departments with a focus on undergraduate teaching. A total of £9.5m was made available centrally, with possibly double that amount being contributed by the universities in terms of staff time, space and facilities, and other resources.

Despite high expectations that the projects' software products would be widely disseminated throughout tertiary education some major difficulties quickly emerged. No attempt was made to standardise hardware platforms and about half of the software was developed on equipment that rapidly became obsolete. There were also other impediments to the intended widespread take-up and portability of the materials throughout disciplines; materials developed in one department were often considered "inappropriate" elsewhere, a phenomenon which became known as the "*not-invented here*" syndrome. With very few exceptions, most of the software products were used solely within the departments in which they were developed. Although the poor level of inter-departmental portability was obviously very disappointing, there were some successes and much was learned from the exercise. Reports on CTI Phase one were produced by Gardner (1988) and Gershuny and Slater (1989).

Table 1

Distribution, by main academic discipline, of projects and funds in CTI Phase 1 (from Gershuny and Slater, 1989)

Academic Discipline	Number of Projects	Mean Funds per Project *UK £ 000s*
Mathematics/Statistics	12	74
Computing/Logic	30	73
Engineering	8	58
Physical Sciences	9	71
Medicine/Biology	14	47
Agriculture/Geography/Geology	12	60
Social Sciences/Humanities	13	49
Law/Business Studies	13	42
Ancient/Modern Languages	9	42
Other Combinations	19	27
ALL DISCIPLINES	139	TOTAL FUNDING £9.5m

(2) CTI Phase 2, 1989-93: Subject-based Centres

The second phase of the initiative was intended to consolidate the successful aspects of the first phase and address areas that required further attention. A more solid infrastructure was to be secured within subject-specific academic domains by providing human support for CBL rather than more software. Competitive bids were again invited and funding was provided to establish 21 subject-based centres throughout the university sector. The subjects covered some 95% of undergraduate teaching areas and a coordinating body, the CTI Support Service (CTISS), was also established. Each centre had the task of promoting the use of CBL in teaching within its individual academic constituency, but was not permitted to undertake software development. Following minor adjustments the initiative now incorporates CTISS and the 20 centres shown in Table 2.

Average annual running costs of each centre are about £40,000 and host institutions also contribute facilities and additional support. Typical staffing comprises a director, coordinator and secretary. Total funding for the second phase has been of the order of about £1m annually, considerably less than for CTI Phase one.

Although the centres' activities vary according to the nature and needs of their disciplines, they all perform an essential core. These include dissemination via print and electronic media of information about good practice in CBL, reviews of software, conferences, workshops and seminars, visits to departments, academic papers, and an advisory service, all available *gratis* throughout the UK university network.

Collectively the centres have been very productive. In the academic year 1992/3, the centres produced 18 subject-based courseware/resource guides averaging 156 pages each. Some 66 Journals and Newsletters were published, and over 6,000 screens of information were posted on academic bulletin-boards maintained by NISS (National Information on Software and Services) and other agencies; 225 software products were reviewed; 23 electronic mail discussion/information lists were set up; 246 departmental visits were conducted, and over 30,500 individual enquiries were dealt with (CTISS, 1994, 5-7).

CTI's second phase has generally been regarded as very successful, bearing in mind the original terms of reference. The centres have been very active and productive, but it has to be recognised that they were never resourced sufficiently to bring about a widespread *cultural* change in the way university teaching was carried out. The human element which the CTI Centres have provided has proved an indispensable form of support without which CBL would remain isolated and perhaps solely the preserve of technology enthusiasts. The need to provide and implement support mechanisms of this type has been widely acknowledged at many levels of the university community, particularly at the "grass roots" department level, and an external audit by French *et al* (1991) commended the centres on both their performance and cost-effectiveness. General reports on the CTI have been produced by Darby (1991; 1992), and Robinson (1992) has reviewed progress within one of the centres.

Table 2
The discipline-based infrastructure of CTI Phase 2

Subject Centre	***Host University***	**Subject Centre**	***Host University***
Accountancy, Finance & Management)	*East Anglia*	Law	*Warwick*
Biology	*Liverpool*	Library & Information Studies	*Loughborough*
Chemistry	*Liverpool*	Mathematics & Statistics	*Birmingham & Glasgow*
Computing	*Ulster*	Medicine	*Bristol*
Economics	*Bristol*	Modern Languages (with Classics)	*Hull*
Engineering	*London (QMWC)*	Music	*Lancaster*
Geography, Geology, Meteorology & Planning	*Leicester*	Physics	*Surrey*
History	*Glasgow*	Psychology	*York*
Human Services	*Southampton*	Sociology, Politics & the Policy Sciences	*Stirling*
Land Use Studies	*Aberdeen*	Textual Studies	*Oxford & Bath*

The CTI model for promoting the use of IT in teaching and learning has attracted great interest in other countries, although little of this interest has yet been translated into action. Through its Council for Renewal of Undergraduate Education, the Swedish government has recently funded a discipline-based initiative based on the UK CTI model. Exchange visits between UK and Swedish CTI staff took place in 1992 and 1993, and a report on this cooperative venture was published recently (Martin, Darby and Kjöllström, 1994). The Republic of Ireland has also recently established a similar organisation. Some 200 academic institutions subscribe to CTI services, and there are many examples of individual CTI Centres having developed informal international associations.

By providing infrastructural support, evaluating courseware and identifying and propagating good practice in its adoption, the CTI Centres have continued their role of enabling a more rapid and comprehensive penetration of CBL into the mainstream of university teaching and learning. However, it was recognised that for fundamental change to occur, CBL had to be integrated into the teaching curriculum on a widespread basis.

The Context of Current Courseware Development

A continued lack of suitable courseware within academic disciplines remained the most pressing need. As a result the CTI recommended that a second programme of courseware development be instituted (Darby, 1991). This recommendation was accepted, but the academic context was radically revised following a major restructuring of organisation of higher education which took place in 1993.

Prior to 1993 the higher education community comprised (a) universities which could award degrees in their own right; (b) polytechnics, which were somewhat more vocationally orientated and whose degrees were validated by the UK Council for National Academic Awards; and (c) other tertiary colleges and institutes. Universities were funded by the Universities Funding Council (UFC) and other institutions were funded by the Polytechnics and Colleges Funding Council (PCFC). On 1 April 1993 the so-called "*binary-divide*" between universities and polytechnics was removed, thereby allowing polytechnics to assume full university status. Thus the number of universities increased from 48 to 115. The UFC and PCFC were replaced by four bodies: separate Higher Education Funding Councils (HEFCs) for each of England, Scotland and Wales, with the Department of Education for Northern Ireland taking responsibility for higher education in that province.

In advance of this major reorganisation the UK government in 1991 initiated policies to increase access to higher education and to double the number of students by the year 2000. However, this was to be brought about *without* a commensurate increase in university resources, which effectively remained static. As a result information technology was perceived by some senior figures in higher education as a means of delivering cost-effective education for increased numbers of students.

(3) The Teaching and Learning Technology Programme

In the 1980s, courseware had been developed piecemeal and the absence of a national strategy for its development and lack of generally-accepted standards made it difficult to integrate such material into existing courses. There was a clear need for courseware that conformed to acceptable standards of design, flexibility, delivery and pedagogy. It was recognised that courseware of this kind is most likely to be produced by nationally-funded consortia of academics collaborating across institutional or departmental boundaries. This strategy was embodied within TLTP which was announced in April 1992. It had a budget of £15m, and would run for a three year period beginning September 1992.

Its objectives were similar to those of the first phase of CTI in that it would support a series of projects each producing courseware for teaching in a variety of disciplines. Bids were invited from both subject-based and institutional consortia as well as individual institutions who could show that materials they produced would be widely adopted (UFC, 1992). In the first phase some 42 projects were approved, bids for 12 of which were coordinated by CTI Centres. The majority of the other 30 were institutional bids. Polytechnics were excluded from participating in the first year because they were not, at that time, funded through UFC: they were allowed to join in the programme in subsequent years however, following removal of the binary divide.

The projects had a wide geographical and disciplinary spread although bids from the natural sciences and engineering outnumbered those from the humanities and social science. The overall character of TLTP projects is also reminiscent of the first phase of CTI. The focus on consortia was intended to minimise the adverse effects of a tradition of local autonomy in curriculum design and delivery which might, despite strong financial control by the central funding body, frustrate the intentions of the initiative that collaboration within disciplines should lead to widespread adoption of the courseware that is developed.

In April 1993 the HEFCs for England, Wales, Scotland and Northern Ireland jointly invited more bids for a second round of TLTP projects with the explicit aim of making teaching and learning more productive and efficient by harnessing modern technology. By the 4 July 1993 deadline 384 bids, for a total of almost £50m had been submitted . Of these, 33 projects were funded, and £3m was committed for the first of three years. In this second phase two CTI Centres, History and Geography, were successful in securing funds: both of them head large consortia of departments within their respective academic communities. As with the first phase of TLTP, the overall aim was to fund the development of transferable, cost-efficient, quality teaching and learning resources which, *inter alia*, facilitate flexible learning strategies, improve course delivery, and make better use of staff time. Productivity gains had to be quantified and attention paid to the issue of common standards of authoring environments. The expectation at the time that the new universities (former polytechnics) would be particularly favoured (as they had been excluded in phase I) proved unfounded. Similarly, an expectation that priority would be given to projects which integrated access to centralised resources, such as data archives and information services, was not matched by the actual allocation of funds. Again science and technology were favoured.

An example of a TLTP project

Finally, by way of illustration, we provide details of the TLTP project with which, as Co-Director and Coordinator, we are closely involved. The project was one of the 33 successful bids under phase II of the TLTP, and commenced in September 1993. It is jointly led by the CTI Centre for Geography and the Division of Geography at Coventry University and involves a consortium of 75 university departments of geography throughout the UK. Resources secured amount to approximately half a million pounds sterling over a three-year period.

The aims of the project, named Geography-CAL, are to specify, develop, test and deliver a library of 21 high-quality transportable CBL modules and other support material intended to facilitate an efficient and effective teaching and learning environment for core topics, concepts and techniques in introductory undergraduate geography courses. Geography, as one of the broadest academic disciplines, draws on material from the natural, physical and earth sciences, the social sciences and the humanities in examining the relationships between people and their environment. Emphasis on analyzing the spatial and temporal patterns of phenomena has meant that graphical skills are fundamental requirements for all its students. Computers are an ideal medium for graphical representation, data analysis and simulation in teaching geography.

The project comprises three main stages: (i) an assessment of academic needs and a review of existing courseware; (ii) development and testing of courseware modules; (iii) an evaluation of the effectiveness and use of modules in the consortium's member institutions. An effective learning environment involves (a) selecting topics, concepts and techniques in which presentation of material by computer is superior to conventional methods of delivery; (b) adopting an interactive problem-orientated approach to learning, which will help develop students' enthusiasm and involvement; (c) developing quality, user-friendly courseware which combines multimedia elements of text, graphics, images, animations, simulations; and (d) formative assessment exercises.

Academic expert panels for each of the modules provide a detailed specification which is then passed to a team of three programmers who will produce the module using authoring software. Because IBM-compatible PCs running under Microsoft Windows represents the most common hardware platform in UK geography departments, Asymetrix' *Toolbook* was selected as the most suitable authoring system.

We envisage that acceptability of the material to university teachers will be enhanced by (a) actively involving potential users in the selection of topics, concepts and techniques and in the development, testing and evaluation of the CAL modules: a programme of workshops for consortium members is being organised from an early stage of the project; (b) circulating a regular newsletter reporting progress on the project; (c) designing each courseware unit around a discrete topic, concept or technique, so as to lend itself to being used independently or in connection with other units; (d) including automated assessment procedures; (e) producing flexible teaching materials suitable for first-year students who have not previously studied geography, whilst allowing those with some previous study of the subject to investigate topics in greater depth; (f) ensuring transportability of courseware by using a common authoring standard; (g) enabling teachers to customise the software to reflect local needs and interests; and (h) providing a staff training and support programme to help potential users to integrate the courseware into their teaching.

Together these mechanisms should increase the feeling of "ownership" of the courseware by potential users and encourage its widespread adoption. The primary efficiency gain will be the saving of time spent in tutorials, practical and fieldwork exercises and assessment; the CAL modules will also replace and complement some formal lecture material.

The project will also be sensitive to the fact that many students will not have an extensive background in geography, many will not initially be well disposed to IT, and that there are gender and other social issues which affect the learning environment. Many courseware modules will typically combine units that link theory with data-handling and other practical exercises to promote the understanding of a clearly-defined set of concepts. In addition to developing new materials, the project will endeavour to integrate existing resources produced under CTI phase I, many of which remain largely undeveloped or unfocused for current needs. Cooperative links have also been agreed and established with various TLTP phase I projects to ensure compatibility of standards and associated integration issues.

Because the project aims to involve virtually all the nation's geography departments, very high efficiency gains are achievable within a short time of the project's completion. First-year teaching — inclusive of lectures, practical classes and tutorials — in some 100 departments involves more than 65,000 hours of staff time annually. If only half of the departments adopt only half of the planned materials, a considerable 4,000 teaching hours could be saved. At a notional £35 per hour this represents £140,000: it is a significant annual return on the investment in the consortium and much higher returns are realistically achievable. Efficiency gains will be focused on several key areas: reduction of routine low-level tasks by production of self-directed learning facilities; minimising repetitive teaching; development of more efficient evaluation procedures; development of student self-assessment procedures; enhanced access to on-line data-sets for teaching purposes; reduction of time spent on producing learning-support materials; more efficient systems for monitoring student progress; more efficient systems of course administration.

As the project has only just begun it is currently too early to assess a measure of success. Our centralised approach — panels of academics passing specifications to professional programmers — is, however, a considered response to the experiences of earlier projects. It was adjudged preferable to the alternative, distributed approach in which individual academics acquire sufficient authoring skills to undertake module production themselves.

Options for the Future

It seems clear that the most appropriate basis for ensuring the widespread implementation of IT-based approaches to teaching and learning in higher education in the UK involves a synergistic and complementary relationship between CTI and TLTP. Following a review of CTI performance in early 1994, the UK HEFCs have all agreed to continue funding the initiative until at least 1999, albeit with some redefinition of roles still to be completed. It is expected that CTI will continue to provide the infrastructure necessary for the successful dissemination of TLTP materials. The traditional focus on support for individual members of academic departments has already been extended by the large increase in subject constituencies following removal of the binary divide. Now CTI is being asked to establish closer links with institutional managements. With the continuing confidence of the HEFCs and the relative security of a prolonged period of funding, the use of CTI to promote IT in higher education is now founded on a far firmer footing than hitherto.

Conclusion

The events and issues described in this paper encompass a series of attempts to establish CBL as a mainstream activity within UK universities. Several of these attempts proved less successful than had initially been anticipated. However, many lessons have been learned throughout the process and it now seems that a successful strategy is now at hand. The key to the successful implementation of computer-based learning clearly entails the integration of hardware, software and "liveware" within a sound and supportive educational environment. This requires change within our institutions that is as much cultural as technological. It is too early to gauge the extent to which TLTP will succeed in achieving its aims. However, it is now accepted that CTI and TLTP are complementary, and indeed both now report to the same advisory group within the Funding Councils. The CTI experience has shown that for any CBL resource to be taken seriously enough to be incorporated into the curriculum it must offer the prospect of being useable for a sufficient length of time to justify the investment of effort involved in its production. It is also clear that effective production of courseware needs collaboration at as many levels as possible: individual, departmental, institutional, discipline, national and even international.

Note: Further information about CTI can be obtained from CTISS, University of Oxford Computing Centre, 13 Banbury Road, Oxford, OX2 6PN, UK. Fax: +44 865 273 275; Email: ctiss@vax.ox.ac.uk

References:

CTISS (1994). *Annual Report of the Computers in Teaching Initiative 1992-3.* Oxford; CTISS Publications.

Darby, J. (1991). Computers in Teaching Initiative: review of the year. *The CTISS File*, 12.

Darby, J. (ed.) (1992). Computers in Teaching Initiative special edition. *Computers in Education*, 19, 1-198.

French, S., Laurillard, D., Makinson, G. & McDonough, R. (1991). *Report on the Computers in Teaching Initiative prepared for the Information Systems Committee of the Universities Funding Council.* Bristol; ISC.

Gardner, N. (1988). Integrating computers into the university curriculum. *Computers in Education*, 12, 23-27.

Gershuny, J. & Slater, J. (1989). *Report on the Computers in Teaching Initiative.* Bath; CTISS Publications.

Martin, J., Darby, J. & Kjöllström, B. (eds.) (1994). *Higher Education 1998 transformed by learning technology.* Lund, Sweden; Council for Renewal of Undergraduate Education, Lund University, and Oxford; CTISS Publications.

Robinson, G. (1992). Activities at the Computers in Teaching Initiative Centre for Geography. *Research and Development in Higher Education*, 15, 276-279.

UFC (1992). *Teaching and Learning Technology Programme.* Bristol; Universities Funding Council Circular 8/92.

Standardizing "HyperVocabulary:" A Proposal

DER-THANQ CHEN
Center for Electronic Studying, College of Education
University of Oregon, Eugene, OR 97403-1215, USA
E-Mail: derthanq@oregon.uoregon.edu

Abstract: Vocabulary used to describe things that are "hyper" is very confusing. This paper discusses four factors which contribute to the confusion. In addition, a set of standardized definitions is proposed. They include: (a) (sequential) text: documents presenting text to be used in a sequential manner, (b) hypertext: documents presenting text to be used in a nonsequential and/or sequential manner, (c) multimedia: documents presenting media to be used in a sequential manner, and (d) hypermedia: documents presenting media to be used in a nonsequential and/or sequential manner. The establishment of these definitions is based on three characteristics of documents: (a) linearity, (b) modality, and (c) singularity. The three characteristics reflect the use of the terms "hyper," "medium," and "multi" respectively. Examples and comparisons of different types of documents are discussed.

The field of hypertext/hypermedia has a communication problem. The vocabulary used to define things that are "hyper" is both confusing and ill-defined. At least four factors contribute to this confusion. They are: (a) similar ideas are described using different terms; (b) the same term is used to describe different ideas; (c) "hyperdocuments" are often confused with electronic documents; and (d) there is inconsistency among definitions of "hypervocabulary". Discussions of each of the factors are presented below.

First, the same idea is often described using different terms. For example, Field (1990) introduced an inexpensive approach to using hypermedia in regular classrooms. In her article, she used the terms "hypermedia" and "interactive multimedia" interchangeably and without defining them. It reads as if multimedia and hypermedia are identical. Similarly, Smith and Westhoff (1992) described the Taliesin Project as a "multimedia project" which includes a "hypermedia delivery system" as the underlying host. They did not, however, clarify the distinction between the terms. Neglecting to provide comparisons among similar terms usually leaves meaning open to interpretation.

Second, even though people sometimes use the same terms, quite often they are referring to different ideas. For example, Conklin (1987) and Marmion (1990) defined hypertext as consisting of nodes of text and links among the nodes. Links are logical connections among nodes. Horney (1991), expanded this node-and-link metaphor a step further. Based on Nelson's (1987) definition, he suggested that hypertext not only includes nodes and links, but also presents information in a way that users are free to choose what to read next. He argued that a key element of hypertext is freedom of choice. Clearly, Horney's use of the term "hypertext" is somewhat different from that of Conklin or Marmion.

Third, people tend to confuse "hyperdocuments" with electronic documents. For example, Bonest (1991) suggested that one of the disadvantages of hypertext is the problem of "tunnel vision;" i.e., readers' vision is limited by the size of the computer screen. However, this is true only if the hypertext system is presented through a computer. Some researchers (e.g., Chen, 1989; Marmion, 1990) have argued that hypertext can also exist on paper. Including tunnel vision as one of the disadvantages of "hyperdocuments" seems to be an overgeneralization.

Fourth, the relationships among definitions of the vocabulary are often inconsistent. For example, Woodhead (1991) suggests that hypertext is a subset of hypermedia, which is a subset of interactive multimedia. He restricted the use of hypertext for text-based documents only. However, later in his book the term multimedia is used to describe documents involving more than one medium. If this is true, how can hypertext be a subset of multimedia? They should be two distinct sets, because hypertext, by Woodhead's definition, includes only one medium, text.

In short, there is a communication problem in this field. I strongly feel the need to propose a set of standardized definitions. In this paper, I define four of the most commonly used terms: (sequential) text, hypertext, hypermedia, and multimedia. By defining these terms, making comparisons among them, and providing real world examples, I hope that we can establish a common ground on which to stand and from which we may communicate more effectively and efficiently.

The Proposed Definitions

The proposed definitions are listed below. Detailed discussion and examples will be presented in succeeding sections. Note that I am using the words "sequential" and "linear" interchangeably. The term "sequential text" is the same as the term "text" in the context of "hypervocabulary."

(Sequential) Text: documents presenting text to be used in a sequential manner.

Hypertext: documents presenting text to be used in a nonsequential and/or sequential manner.

Multimedia: documents presenting media to be used in a sequential manner.

Hypermedia: documents presenting media to be used in a nonsequential and/or sequential manner.

Here, I dedicate the definitions of "hypervocabulary" to describing types of documents. These definitions do not include the programs or systems which hold or produce the documents. Nor do they refer to constructs such as nonlinearity or to the manner in which the documents are used. When referring to these systems, I will use "hypertext systems" or "multimedia systems." When referring to constructs, I will use the "idea" or "concept" of hypermedia. Therefore, "hypertext documents" means the same as "hypertext" based on the proposed definitions. Furthermore, the word "document" is taken as a token which means any container of information. A document can be either electronic or nonelectronic.

Additionally, I would like to distinguish between an author's intention and the user's reading strategies. It is the author's intention, not the user's, which determines the characteristics (e.g., linear or nonlinear) of a document. Any document can be linear or nonlinear, if the definitions are based on the reader's use. For example, a naive reader might decide to read an encyclopedia from the first page to the last, even though the encyclopedia was originally designed to be read nonlinearly. Similarly, an expert reader usually reads a book nonlinearly by referring back and forth among different pages, even though a book was usually written to be read linearly. To avoid unnecessary confusion, the proposed definitions are based on how documents are designed to be read, but not how users read. More discussions will be presented later.

Linearity, Modality and Singularity

To discuss the proposed definitions, I have chosen to start from describing three characteristics of documents: linearity, modality, and singularity. The three characteristics are meant to reflect ideas of "hyper," "medium," and "multi" respectively. Their relationships are shown in Figure 1.

Linearity indicates whether or not a document is organized in a linear manner. Conventional books, for example, were written with the expectation that readers read them in a linear manner. Mystery books, in particular, do not expect readers to read the ending first (although many of them do). It is expected that readers will enjoy guessing the ending by reading the books linearly. On the other hand, a dictionary, is designed to be read in a nonlinear manner. The definition of each word in a dictionary can be considered a node. Although these nodes are in an alphabetical order, users of a dictionary do not read the definitions in an alphabetic, i.e. linear, order. Users of a dictionary look up words by going directly to a particular page at the appropriate spot in the alphabet. They do *not* read the dictionary from the beginning to the end. Furthermore, when encountering an unfamiliar word in a definition, users often go to the page that defines the unfamiliar word. An experienced dictionary user usually jumps back and forth among several pages to check the meaning of different words. In this case, a dictionary is designed to be used nonlinearly, and the actual links are constructed by the reader in real time.

Modality characterizes the type of media included in a document. For example, a document of sounds has a different modality from a document of graphics. A document with video has a different modality from a document without video. As will be discussed later, modality, in this proposal, is used to distinguish documents with text from documents without text.

Singularity describes the number of media involved in a document. If a document includes only one medium, it is singular. If more than one medium (at least two) is included in a document, it is not singular. Thus, singularity can be used to describe the distinction between text and multimedia. Text includes only one medium, whereas multimedia implies the involvement of mixed media such as sounds and pictures.

Among these three characteristics, the concept of linearity is the most controversial. When defining the terms hypertext and hypermedia, people tend to emphasize their nonlinearity, giving the impression that nonlinearity is the only characteristic of "hyperdocuments." In fact, "hyperdocuments" include nonlinear as well as linear characteristics. This is very clearly indicated by Nelson (1987), when he says that "hypertext can include sequential text, and is thus the most general form of writing" (p. 0/3). In other words, hypertext is the union of linear text and nonlinear text.

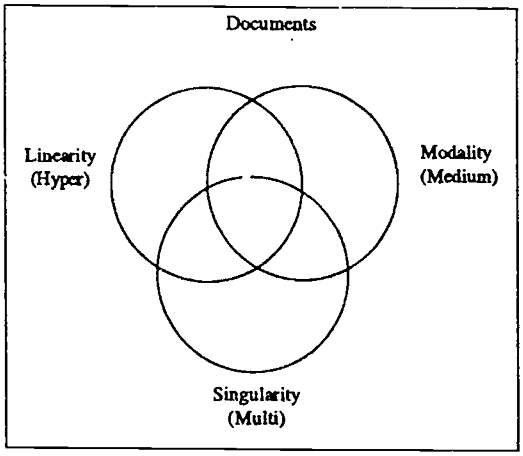


Figure 1. Linearity, modality and singularity characteristics of documents

A good example of this relationship is provided in the book *Literary Machines* (Nelson, 1987). *Literary Machines* is organized into one Chapter Zero, several Chapters One, one Chapter Two, several Chapters Three, several Chapters Four, and several Chapters Five. Nelson suggests that the reader read Chapter Zero and one of the Chapters One, and then Chapter Two, the heart of the book. "Because Chapter Two is long and sequential, its parts are numbered. Other sections of this book are not numbered because they are not, in principle, sequential" (p. 0/3). Nelson suggests that the reader then read one of the closing chapters. Chapters Four and Chapters Five are particularly nonsequential; readers are encouraged to read in whatever order they prefer. Readers are also encouraged to read the book several times, taking a different path each time. *Literary Machines* exemplifies both the linear and nonlinear characteristics of hypertext, and provides a concrete example that hypertext is a superset of sequential text.

In the same way that hypertext is the most general form of writing, hypermedia can be considered the most general form of media. That is, hypermedia can include sequential media and nonsequential media. In other words, hypermedia is the union of linear media and nonlinear media. Because the linguistic structure of the word "multimedia" does not imply the possession of nonlinear characteristics, it is best used to describe linear media only. Therefore, multimedia is a subset of hypermedia.

A Closer Examination

Using the characteristics of linearity, modality and singularity, eight types of documents can be derived. They include: (a) nonlinear nontextual medium, (b) linear textual medium, (c) linear media without text, (d) nonlinear textual medium, (e) nonlinear media without text, (f) linear media with text, (g) nonlinear media with text, and (h) linear nontextual medium. Their relationships are illustrated in Figure 2.

Area A is nonlinear nontextual medium. For example, the program "Inigo Gets Out" (Goodenough, 1987) is a pictorial story presented in a nonlinear manner. The main character "Inigo" of the story is a cat. The reader plays the role of Inigo and decides what to go next throughout the journey of its adventure. The reader can literally read the story several times without repeating the same path. "Inigo Gets Out" is nonlinear and nontextual, and singular, involving only the medium of picture. There is currently no accepted term for a document with these characteristics. To be consistent, the term "hypermedium" is a possible descriptor.

Area B is linear textual medium. As mentioned above, a mystery book involves only the text medium and is presented in a linear fashion. It is textual and singular. Therefore, it falls into this area. A document with these characteristics should be called "text."

Area C is linear media without text. Although most existing documents involve text, there are some that do not have text in them. For example, a video tape program usually involves sounds and moving pictures. But the users (audiences) normally only access it in a sequential manner, playing it from the beginning to the end. It is linear and not singular. No text is involved. A document with these characteristics should be called "multimedia."

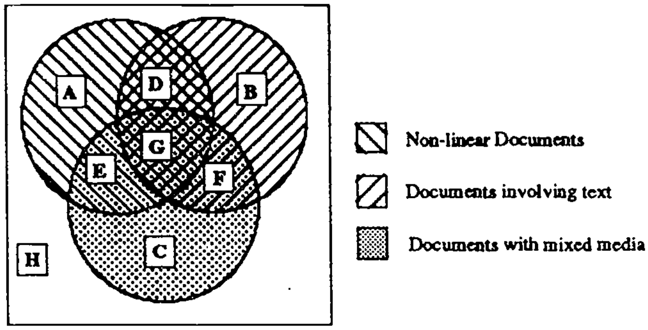


Figure 2. Eight types of documents derived from the three characteristics

Area D is nonlinear textual medium. Again, as mentioned above, a dictionary is nonlinear and is mainly textual. Only one single medium, text, is involved. A document with these characteristics should be called "hypertext."

Area E is nonlinear media without text. For example, when a video program is divided into segments and is put on a laserdisc, it becomes accessible by the user in a nonsequential manner. The user can choose which segment and which order of video clips to examine. No text is involved. It is nonlinear and not singular (includes both motion pictures and sounds). A document with these characteristics should be called "hypermedia."

Area F is linear media with text. When watching a foreign movie, we may need the text captions to understand the film. A captioned film is still linear, but is not singular. It now includes text. A document with these characteristics should also be called "multimedia."

Area G is nonlinear media with text. For example, a CD-ROM encyclopedia falls into this category. An electronic encyclopedia usually has text, sounds and pictures. It can be accessed in many possible orders. One of the media involved is text. Therefore, it is nonlinear, not singular and involving text. A document with these characteristics should also be called "hypermedia."

Area H is linear nontextual medium. An audio type, for example, falls into this area. Music recorded in a type is to be listened linearly. It involves only one single medium, sound and it is nontextual. There is currently no accepted term for a document with these characteristics. To be consistent, the term "monomedium" is a possible descriptor.

Table 1 summarizes characteristics and examples of different tapes of documents.

Summary

In summary, based on the above discussion, the proposed definitions for a "hypervocabulary" convey the following statements:

- Text, hypertext, hypermedia, and multimedia are types of documents.
- Hypertext and hypermedia exist in both electronic and other media forms.
- Hypertext is a superset of text.
- Hypermedia is a superset of multimedia.
- Text and multimedia are two distinct sets.
- Hypermedia and hypertext are two distinct sets.

The relationships among text, hypertext, multimedia and hypermedia can be represented in Figure 3. A simplified version of this figure is shown in Figure 4.

One of the things I like about the proposed definitions is that now we are ready to invent new terms without adding confusion to "hypervocabulary." The relationships presented in Figure 2 apply to different media.

Theoretically, we shall have "hypersound," "hypervideo," "hyperpicture," and so on. Any combination of the above (including hypertext) will make a document hypermedia. How to draw the relationships among these terms, as a general strategy used by many textbooks, will be left as an exercise for the readers.

Table 1. Summary of types of documents in Figure 2

Area	Linear	Textual	Singular	Description	Example	Vocabulary
A	No	No	Yes	Nonlinear nontextual medium	"Inigo Gets Out"	Hypermedium
B	Yes	Yes,	Yes	Linear textual medium	Mystery books	Text
C	Yes	No	No	Linear media without text	Video without text caption	Multimedia
D	No	Yes	Yes	Nonlinear textual medium	Dictionaries	Hypertext
E	No	No	No	Nonlinear media without text	Laserdiscs of sounds and pictures	Hypermedia
F	Yes	Yes	No	Linear media with text	Video with text caption	Multimedia
G	No	Yes	No	Nonlinear media with text	CD-ROM encyclopedia	Hypermedia
H	Yes	No	Yes	Linear nontextual medium	Audio tapes	Monomedium

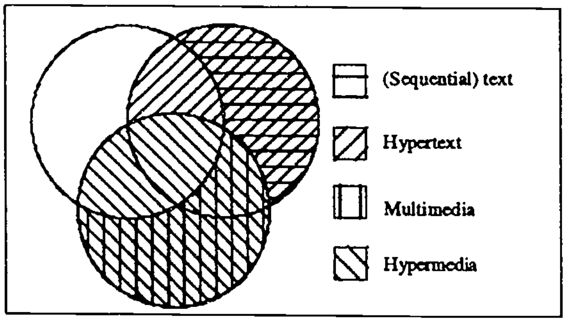


Figure 3. Relationships among text, hypertext, multimedia and hypermedia

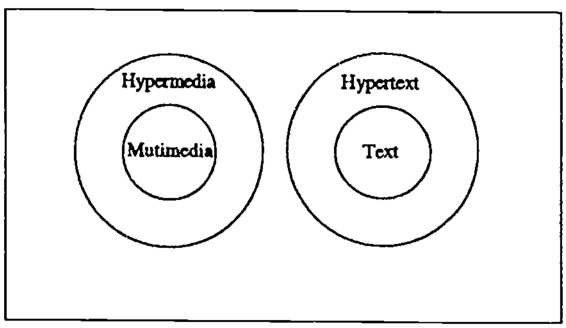


Figure 4. A simplified version of Figure 3

References

Bonest, I. D. (November, 1991). An alternative approach to hypertext. *Educational and Training Technology International, 28*(4), 341-346.

Chen, C. (1989). As we may think: thriving in the HyperWeb environment. *Microcomputers for Information Management, 6*(2), 77-79, 81-97.

Conklin, J. (1987, September). Hypertext: an introduction and survey. *Computer*, 17-41.

Field, C. E. (1990). Exploring hypermedia. *inCider, 8*(1),36-44.

Goodenough, A. (1987). *Amanda stories* [Computer program]. Santa Monica, CA: Voyager.

Hanfling, S. S. (1990, February). *Hypermedia and instruction: Remembrance of lessons past*. Paper presented at the AECT National Convention Workshop, Alexandria, VA.
Horney, M. A. (1991). *Case studies of authoring in hypertext*. Unpublished doctoral dissertation, University of Oregon, Eugene, OR.
Marmion, D. (1990, June). Hypertext link to the future. *Computers in Library, 10*(6), 7-9.
Nelson, T. H. (1965). A File Structure for the Complex, the Changing and the Indeterminate. In *Association for Computing Machinery Proceedings of the 20th National Conference* (pp. 84-100). New York: Association for Computing Machinery.
Nelson, T. H. (1987). *Literary Machine*. South Bend, IN: The Distributors.
Smith, E. E. & Westhoff, G. M. (1992, January). The Taliensin project: multidisciplinary education and multimedia. *Educational Technology, 32*(1),15-23.
Woodhead, N. (1991). *Hypertext and Hypermedia: Theory and Applications*. Wilmslow, England: Sigma.

CONFERENCE ON A DISK:
A Successful Experiment in Hypermedia Publishing
(Extended Abstract)

M. CHEYNEY[1], P. GLOOR[2], D.B. JOHNSON[1], F. MAKEDON[1], J. MATTHEWS[1] AND P. METAXAS[3]

[1] *Department of Mathematics and Computer Science*
Dartmouth College, 6211 Sudikoff Laboratory, Hanover, NH 03755, USA
E-Mail: {cheyney,djohnson,makedon,James.W.Matthews}@dartmouth.edu

[2] *Union Bank of Switzerland*
8021 Zurich, Bahnofstrasse 45, Switzerland
E-Mail: Peter.Gloor@zh014.ubs.ubs.ch

[3] *Department of Computer Science*
Wellesley College, 106 Central Street, Wellesley, MA 02181, USA
E-Mail: PMetaxas@lucy.wellesley.edu

Abstract: Academic conferences are a long-standing and effective form of multimedia communication. Conference participants can transmit and receive information through sight, speech, gesture, text and touch. This same-time, same-place communication is sufficiently valuable to justify large investments in time and travel funds. Printed conference proceedings are attempts to recapture the value of a live conference, but they are limited by a fragmented and inefficient approach to the problem. We addressed this problem in the multimedia proceedings of the DAGS'92 conference. The recently published CD-ROM delivers text, graphic, audio, and video information as an integrated whole, with extensive provisions for random access and hypermedia linking. We believe that this project provides a model for future conference publications and highlights some of the research issues that must be resolved before similar publications can be quickly and inexpensively produced.

Introduction

Academic conferences are a long-standing and effective form of multimedia communication. Conference participants can transmit and receive information through sight, speech, gesture, text and touch. This same-time, same-place communication is sufficiently valuable to justify large investments in time and travel funds. Traditional printed conference proceedings and session videotapes are attempts to recapture the value of a live conference, but they are limited by a fragmented and inefficient approach to the problem. Video and audio tapes require as much time as the recorded events, while printed proceedings can not convey the dynamic nature of a conference. But more importantly, they are not integrated with each other.

In preparing the DAGS'92 Multimedia Conference Proceedings we aimed to address these shortcomings, and therefore deliver more of the value of an academic conference to our audience.

Recently, conference organizers have realized the endless possibilities that the multimedia productions can have, and some first efforts on multimedia proceedings have already been published (Rada, 1993; MacSciTech, 1992). Even though these are efforts to the right direction, we believe that they fall short of the multimedia abilities.

The DAGS'92 CD-ROM (Gloor, Makedon & Matthews, 1993) delivers text, graphic, audio, and video information as an integrated whole, with extensive provisions for random access and hypermedia linking. We believe that this project provides a model for future conference publications and highlights some of the research issues that must be resolved before similar publications can be quickly and inexpensively produced. The experience gained from this effort will be applied not only to multimedia conference proceedings of the future, but also to multimedia textbooks and learning environments. We view this as the strongest point of our experiment and research, and a direction that should be pursued further.

Background

In June 1992 Dartmouth College hosted the first annual Dartmouth Institute for Advanced Graduate Studies (DAGS) symposium, on the topic of "Issues and Obstacles in the Practical Implementation of Parallel Algorithms and the Use of Parallel Machines." The symposium program consisted of eight talks by invited speakers plus thirteen contributed talks, presenting a total of twenty-two papers (one invited talk spanned two papers). Since the topic is considered the central problem in the area of parallel computation today, it was the intention of the organizers to make the results as widely available and accessible as possible to the parallel computing research and teaching communities. In addition to the usual printed Proceedings (Johnson, Makedon & Metaxas, 1992), it was decided to publish multimedia proceedings that would capture as much as possible of the conference atmosphere.

Description of the System

The multimedia proceedings on CD-ROM we produced delivers the twenty-two papers in hypertext integrated with the eight invited talks in Apple's QuickTime movies. Hypermedia links and random-access are used extensively in this integration.

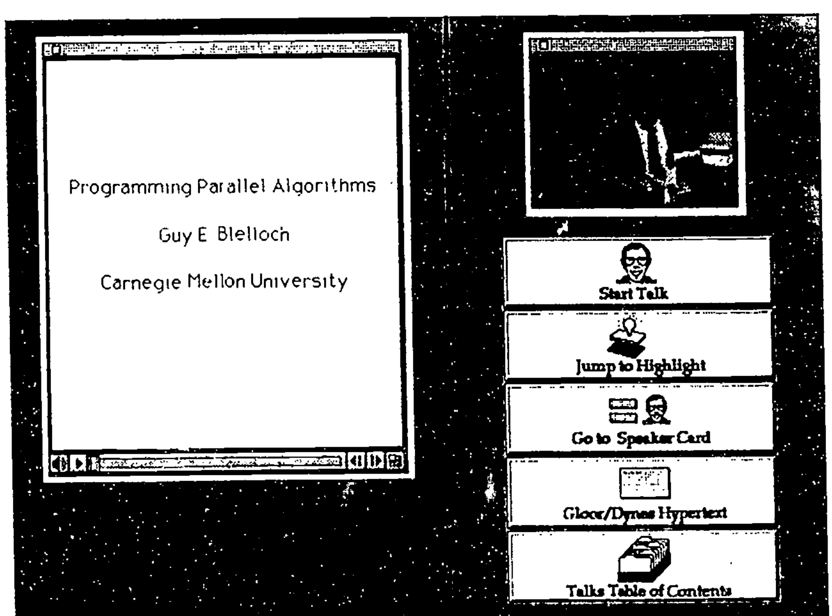


Figure 1: The Talks interface.

The CD-ROM includes: A navigation shell that facilitates hierarchical navigation in hyperspace, hypertext of the papers presented, movies of the invited speaker delivering the conference talks along with their slides being marked during the talk, hyperlinks connecting relevant parts of the proceedings, and bibliography. In the future, it would be nice to include animations of some of the algorithms presented. Furthermore, the system is extendible in the sense that the user is able to create his/her own hyperlinks among objects that he/she deems relevant, do search on keywords and keep notes on the documents.

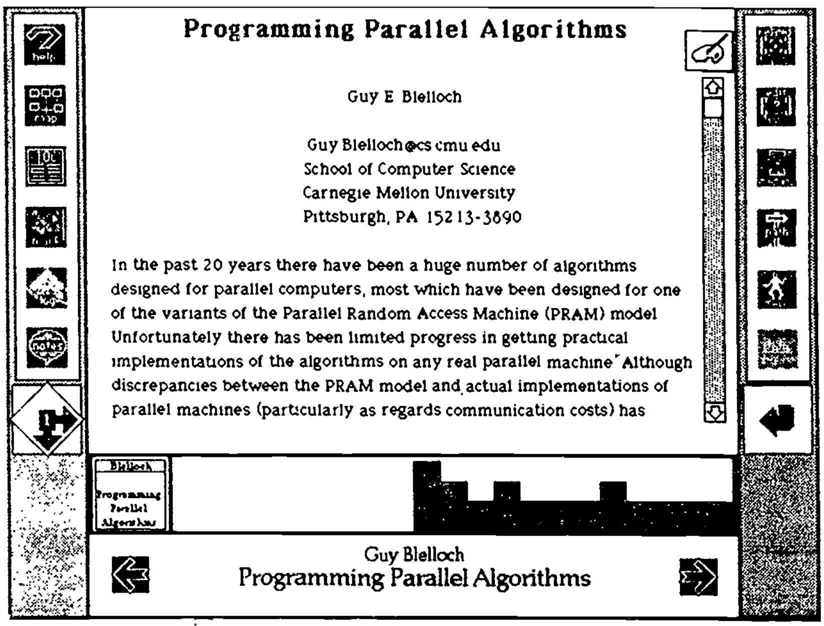


Figure 2: The hypertext interface.

In a typical session, a user using our system could first get a quick overview of the contents of the CD-ROM. Then he/she could follow a talk on a particular topic that seems interesting, by watching the movies of the slides and of the speaker in separate windows. The user could also get an overview of the talk by using the pop-up menu containing the section titles of the talk, or by skimming though the slides. If, at some particular point, the speaker mentions a theorem without proof, as is often the case in conferences, the user of our system could jump into the hypertext to read the omitted theorem proof in detail. Then, he/she could go back to the talk, continue reading the hypertext bringing on the screen several windows containing relevant information, or do a search on some keyword to find out who else mentioned this keyword during the conference. Assuming that the search brought up several candidate sections of papers, the user could jump into the new paper and continue reading from there, or even jump into the video movie of the second speaker and see how the material was presented during that talk. In every place, the user could make his/her own remarks on the subject being read and keep them for later examination or for filling in the background gaps.

There is considerable flexibility built in our system; evaluating which parts are crucial and should be kept and enhanced, and which could be dispensed without affecting the overall performance of the multimedia proceedings, is part of future research and evaluation.

Development Stages

The first step in producing the multimedia proceedings was to collect the raw material. The speakers' presentations were videotaped, their overhead slides were copied, and their papers were collected. To make the synchronization easier, we used two videocameras for the videotaping; one focusing on the speaker and one on the projected slides. It turned out that the second was by far the most useful. After the collection, all these materials were converted to digital form for computer-based processing. Due to the space constraints of CD-ROM, our intended delivery medium, we decided to deliver only the eight invited talks in a full audio/visual form.

Despite the huge storage provided by the Compact Disk technology we knew that we could not fit even the eight one-and-a-half hour video tracks on a CD-ROM. Therefore we decided to display a one-minute video loop of each speaker. This was certainly the right decision because little valuable information was lost this way; it turns out that a minute of video is sufficient to convey a sense of a speaker's appearance and mannerisms, while at the same time is less distracting to the person studying the talk. Furthermore, breaking the synchronization between the audio and video tracks allowed us to edit the audio without introducing video skipping. Editing both the audio and the video data of the speakers, in a way that preserves synchronization and smooth transition between edited frames, seems as a formidable task which we did not have the tools to undertake.

The audio track of each talk was digitized and then edited to remove pauses and noise words such as "umm"s and "ahh"s. The edited talks were roughly half as long as the originals, and much more listenable. To improve the quality, we amplified most of them using a commercial sound processing application.

The overhead slides were scanned and edited for clarity and contrast; they were also made smaller to fit in the appropriate window of the interface program. One of the problems we encountered in this process was the fact that, after diminution, the slides that were poorly handwritten had low readability. So, some of the slides had to be retyped.

Then, using the original videotapes as guides, we synchronized the slides to the edited audio tracks using a commercial video editor. The resulting "movie" reproduces the most important features of a talk, the speaker's words and slides, and preserves their temporal connection. These "movies" were indexed to allow random access to a list of primary topics, and to allow more sensitive linking between the papers and the talks.

The video loops were digitized with a low-end video capture board and compressed to provide efficient playback from a CD-ROM. The loops were selected to have similar opening and closing frames, so that the loop transition would not be distracting to viewers. The loops were kept small to enhance playback performance and to keep the lack of synchronization between the speaker's lips and words from becoming a distraction.

Given the variety of the playback speeds of the commercial CD-ROM drivers available, some computer configurations will have a hard time displaying both the slides video and the speakers video loops on the screen at a comfortable speed. For that reason, we have given to the user the ability to stop the speaker video loop and replace it with a static color picture of the speaker.

The final, and by far the most time-consuming stage, was to prepare the twenty-two papers. Wanting to produce a truly hypermedia product and not to sacrifice its usability, we decided to present the papers in a hypertext form, using an advanced hypertext engine. We used the Gloor/Dynes hypertext system (Gloor, Dynes & Lee, 1993) that was developed at MIT for the CD-ROM version of the "Introduction to Algorithms" textbook (Cormen, Leiserson & Rivest, 1990) and is based on Apple's HyperCard software.

We first broke the twenty-two papers into hypertext "nodes," and assigned each node a "node level" that reflected its degree of generality. For example, all the abstracts of the papers are on level 1. This way, a user can quickly become familiar with the themes of the presented papers by visiting all the first level nodes of the Proceedings. This "chunking" process was handled manually by computer scientists with expertise in the subject area. It appears that one cannot automate this process unless the authors have written their papers following some carefully predefined specifications. We also tried to have authors provide chunking information, but the results were not sufficiently consistent from paper to paper, so in the end all chunking was performed at Dartmouth.

We, then, converted all the papers to HyperCard text. Every author provided postscript versions of the papers and half of them provided electronic versions in TeX or LaTeX format; the latter were converted to HyperCard form with homemade utilities and manual cleanup. Unfortunately, we could not use postscript files for the hypertext engine, so the remaining papers were scanned, processed by optical character recognition software, and then manually corrected. A great number of errors were introduced by the scanning and recognition process, and some pages were simply retyped. A number of text features required special treatment. HyperCard does not support subscripts or superscripts, so, special fonts were used in their place. Uncommon symbols were similarly provided by custom fonts. This still leaves out very complex equations, which were scanned and displayed as graphics using custom software.

Figures were scanned and edited for clarity. The text was manually marked to provide links to citations, tables, and figures. Each hyperlink in the text appear as bold-faced word. The system supports multiple windows containing scanned figures, tables or bibliographic information. Finally, hyperlinks leading to referenced sections were introduced.

The final step was to integrate all these elements into a single user interface, and to test the resulting system. The interface was designed to be simple, usable and attractive. Extensive color graphics and on-line help facilities were built into the navigation shell. The system was tested on a number of machines with different capabilities and configurations. Special care was taken to optimize the transfer data ratio so it performs well on a variety of commercially available CD-ROMs.

Research Issues

The production of the DAGS'92 Multimedia Proceedings has confronted us with two subjects for future research. One is the question of how such projects can be assembled in a short period of time, with less manual labor. Even though, in the beginning, some production delays were caused by evaluation of the best alternative choice in problems we faced, the experience gained has resolved most of these problems and we now need to focus on the automation of the manual steps.

Some of the time-consuming steps, such as digitizing papers and slides, could be eliminated if the source material was available in electronic form. Removing pauses and noise words from audio sources automatically should also be possible, with sufficiently sophisticated techniques. We currently experiment with more advanced software that can help in that direction (Matthews, Gloor & Makedon, 1993)

Converting linear text to a hierarchy of hypertext nodes may be the most intractable problem; it is difficult to see how to replace the human expert, in the near term. It is clear that projects such as this will not be undertaken if they require the multiple man-years of labor we invested; so progress must be made in automating the process. We are currently working on a carefully specified list of rules and instructions that could help the authors divide an article in hypernodes. Ultimately, we hope that they would use this list in preparing the final version of the paper. This approach, if successful, will eliminate the most difficult problem of future productions.

The second subject for research concerns the effectiveness of the final result. We believe that the multimedia proceedings provide most of the content of live talks and linear papers, with the advantage of random access and hypertext linking. Of particular interest is using this technology in the production of hypermedia books. It is often the case that a good textbook author is also a good speaker; hypermedia textbooks, based on the results of our experiment, are expected to have positive impact on the writing of future textbooks. Further exploration of this is very important. But this proposition needs careful evaluation, and the cost/benefit of certain features (such as video loops and hypertext) needs to be scrutinized in order to justify the production resources they require.

Conclusions

We believe that the DAGS'92 Multimedia Proceedings is a step towards academic publications that more fully reproduce the experience of a live conference or classroom. Currently such publications are one of a kind, expensive to produce, and with clear but unmeasured advantages over their traditional counterparts. Our experience highlights these shortcomings, but also suggests that with further research improved systems can be built with less effort, and greater rewards.

References

Rada, R. (1993) (prod. chair) *Proceedings CD-ROM of the First ACM International Conference on Multimedia*, 1-6 August, Anaheim, California.

MacSciTech (1992). *Proceedings CD-ROM of the 1992 MacSciTech Conference on Scientific and Engineering Applications for the Macintosh*, 15-17 January, San Francisco, California.

Johnson, D.B., Makedon, F. & Metaxas P. (1992). (eds.) *Proceedings of the 1992 DAGS Conference*, 23-27 June, Hanover, New Hampshire.

Gloor, P., Dynes, S. & Lee, I. (1993). *Animated Algorithms*, MIT Press, Cambridge, Massachussetts.

Cormen, T., Leiserson, C. & Rivest, R. (1990). *Introduction to Algorithms*, MIT Press, Cambridge, Massachussetts.

Matthews, J., Gloor, P. & Makedon, F. (1993). VideoScheme: A Programmable Video Editing System for Automation and Media Recognition. In *Proceedings of the ACM Multimedia '93*, Anaheim, California.

Gloor, P., Makedon, F. & Matthews, J. (1993) (eds.) *Multimedia Proceedings of the DAGS '92 Conference*, TELOS/Springer-Verlag.

Assessing the Effectiveness of Multimedia in Language Learning Software

DOROTHY M. CHUN
Dept. of Germanic, Slavic and Semitic Studies
University of California, Santa Barbara, CA 93106-4130, U.S.A.
E-mail: dmchun@humanitas.ucsb.edu

JAN L. PLASS
Fakultät Informatik
Technische Universität Dresden, 01069 Dresden, Germany
E-mail: jp3@irz.inf.tu-dresden.de

Abstract: In this paper, we assess the effectiveness of a multimedia program for reading authentic German texts in three areas. First, based on user evaluation of the visual interface design, we assessed the *usability* of the program with particular regard to user reaction to the multimedia components of the program. Secondly, we tracked *learner behavior* while using the program and recorded the types of multimedia links that were chosen. Thirdly, we assessed the *effectiveness* of the program for two levels of *language learning*: short-term recall of vocabulary items and overall reading comprehension. Based on answers to questionnaires, users felt overwhelmingly that seeing a photograph or movie helped reinforce learning but that the text and audio links were not as helpful. The user logs indicated that a greater proportion of picture and movie links were chosen than text links. The vocabulary tests show that vocabulary words were learned and recalled better when "defined" with picture or movie links than when only textual definitions were available.

Based on current research and theories of reading comprehension, in order for understanding and learning to take place, a reader must first activate prior knowledge, must then reorganize these existing knowledge structures or schemata, and then can link them to the new material to be learned (cf. Carlson 1990, Bernhardt 1991). Our multimedia program for the Macintosh computer aids in this process, first, by providing a contextualized overview or preview of each reading text in the form of a QuickTime movie clip, which is designed to activate in the learner's mind the text's global concepts. Second, by avoiding simple translations of unknown words and phrases in the text and instead providing other types of information in the form of hypermedia links, such as graphics (pictures), video (QT movies) and sound, in addition to the definitions of words, the learner is encouraged to infer the meaning of the word or phrase. These integrated types of information help readers formulate their own cognitive schemata, and research has shown that students learn more when they "make their own connections" rather than having connections made for them (or in this case, having translations given to them, cf. Hulstijn 1992). The interactive nature of the software allows learners to take an active role in the process, i.e. they can choose the type of information or media link that is most helpful to them. At the same time, the program provides individualized options and self-pacing ability, in order to accommodate different learning styles. In addition, increasing students' motivation to learn is the goal of any software but one which multimedia applications seem particularly successful at because they address multiple senses and thus reinforce learning.

Key Program Features of *CyberBuch*

CyberBuch is a multimedia program whose prime instructional goal is to improve the comprehension skills of students who are beginning to read authentic German texts. The metaphor of a library introduces the user to the program, and from there, the following key features are accessible:

1. Vorschau "Preview": This component is an advance organizer, a means of conveying global information about the text, activating top-down processing strategies of first understanding macro-level ideas before micro-level details.

2. Native Speaker Reading the Text ("Text-Movie Link"): Accompanying the written text is the option of hearing the text read by native speakers. Students may listen to the entire text or select any sentence they want to hear. They control the playback as one would a tape.

3. Multiple Media Links: Within the text, certain words are marked as "hotspots" or "hot text" and are linked to various kinds of information in the form of text, still graphics or videos. For each word, up to three different links are available. These "multiple media links" are accessed by dragging the word and dropping it on one of the three media icons. In addition to the media link selected, the word is also spoken as a default.

Effectiveness Testing

Researchers today have moved away from the traditional comparative research (medium A vs. medium B, CALL (computer-assisted language learning) vs. non-CALL). Rather, more descriptive and evaluative studies are being conducted, among other purposes, to acquire "new theoretical information about the psycholinguistic nature of language learning and the way in which a specific computer capability of delivering instruction –its coding elements– has been shown to affect or interact with the learning process (Pederson 1987). As Clark (1983) sums it up, "We have moved from asking which medium was a better teacher to a concern with which 'attributes' of media might combine with learner traits under different task conditions and performance demands to produce different kinds of learning."

The *CyberBuch* program was pilot-tested by 44 second-year German students at the University of California, Santa Barbara in November 1993. We tested three separate but related aspects of the program and its effectiveness: *usability, learning behavior, effects on language learning.*

1. Usability: We first assessed the *usability* of the program, with particular regard to the multimedia components of the program based on questionnaires students filled out after they had worked with the program for two 50-minute class periods. There were a total of 42 questions, ranging from ease of use of the program, to the clarity and effectiveness of the multimedia links, and the users' perceptions of their own learning styles. The most relevant questions for this paper involve student opinions of the various links and perceptions about how they learn best.

Figure 1 shows that an overwhelming percentage (95% and 92%, respectively) agreed or strongly agreed that "The still images [picture links] clearly showed what the word means" and that "The movies clearly showed what the word means." The reactions to the statement "The German definitions [text links] for the word were clear and understandable" were mixed, with roughly a third (30%) agreeing or strongly agreeing, a third (32%) uncertain and a third (35%) disagreeing or strongly disagreeing. However, according to students' verbal and written comments beyond the questionnaire, this is due *not* to the type of link, i.e. textual, but rather to the difficulty of understanding definitions in German.

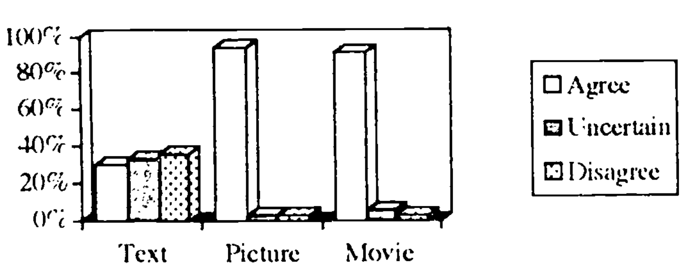


Fig. 1. Clarity of links

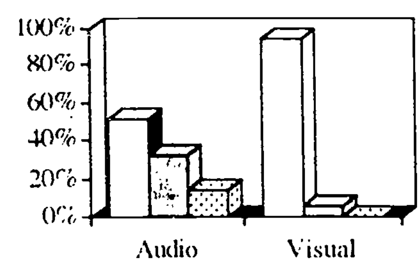


Fig. 2. Helpfulness of links for learning

Similarly, Figure 2 shows that an overwhelming percentage (95%) agreed or strongly agreed that "Seeing a photograph or movie [visual link] helped to reinforce learning." However, only about half (51%) agreed or strongly agreed that "Hearing the words spoken [audio link] helped to reinforce learning."

Students were also asked to rank the different media links in order of which helped them to learn remember individual vocabulary words best, with 1 = helps most, 4 = helps least. The striking results are that nearly half

(43%) of the students ranked pictures as helping the most and the other half (46%) ranked movies as helping the most, while 51% ranked textual definitions as the third most helpful link and 68% ranked hearing the word spoken as the fourth most useful link (see Figures 3 and 4).

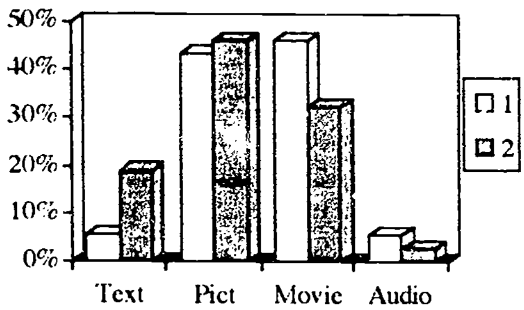


Fig. 3. Ranking of most useful links

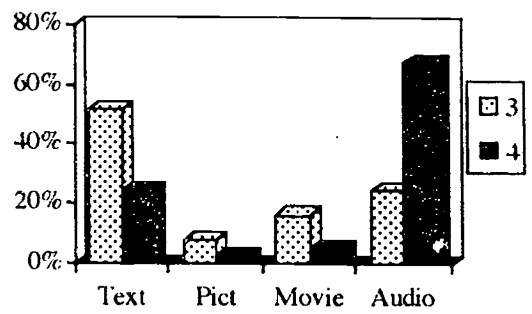


Fig. 4. Ranking of least useful links

2. Learner Behavior: Secondly, we tracked *learner behavior* while they used the program, hypothesizing that the picture and movie links would appeal to students and they would choose such links to a greater degree than "typical" textual or definitional links. The logs recorded provide data about which links were chosen and which ones were not used at all for both days that the program was used.

After learners had used the program for two class periods, a "surprise" vocabulary test was given, consisting of 17 words. The words were selected to represent the different multimedia links: 6 of them had only text links (definitions in German), 5 had only picture + text links, 5 had only movie + text links, and 1 had all three links (picture + movie + text). It should be noted that the audio component was included by default with *all* of the links. In this paper, we are restricting ourselves to the log data from these 17 words (and excluding the other 59 words in the program with links).

The data show that when text links were available (17 words x 44 students = 748 available links), 38% of these links were chosen. While there were fewer available picture and movie links (6 x 44 = 264 for each type), they were chosen to a much greater degree, 58% and 74%, respectively.

Arrow shows number of available links for each media type.

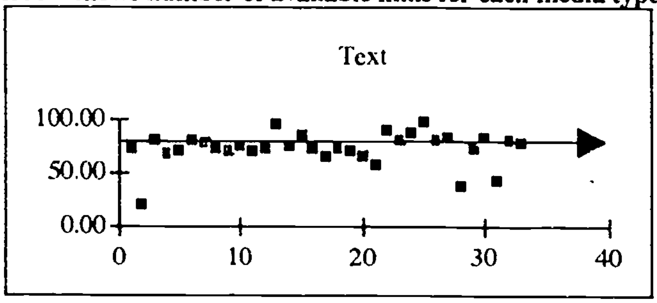


Fig. 5. Distribution of text links chosen by individual students

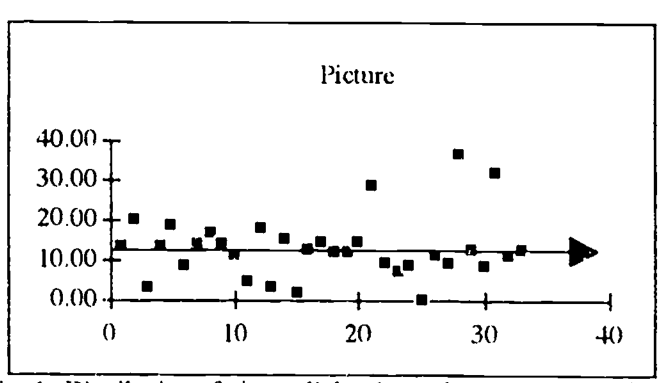


Fig. 6. Distribution of picture links chosen by individual students

Figures 5-7 show the distribution of all the media links chosen by individual students, including multiple selection of a link. The arrows show the number (percentage) of available links for each type.

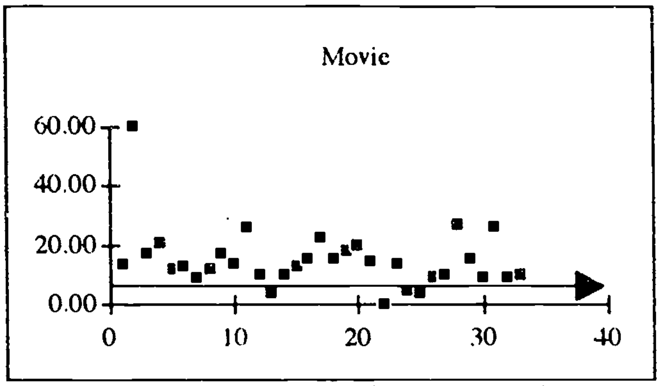


Fig. 7. Distribution of movie links chosen by individual students

In addition to tracking the chosen links for these 17 words, we asked learners on their vocabulary quizzes to specify which link they thought helped them learn the word. We then compared these "self-reported" data to the logs which recorded actual behavior and found a significant dependence between the type of link chosen and whether learners reported that that particular type of link was useful. For the text links, there was a negative relationship (Pearson χ^2 (1) = 51.67, $p < 0.05$), i.e., when text links were chosen, students named a *non*-text link as being most helpful in 53% of the cases. In less than half of the cases (47%) a text link was found to be useful. On the other hand, for the picture and movie links, there was a positive relationship for both types of links between the actual choice of a link and reporting that it was useful. For picture links, Pearson χ^2 (1) = 20.26, $p < 0.05$, i.e., when picture links were chosen, students reported that the picture link *was* helpful in 57% of the cases; for movie links, Pearson χ^2 (1) = 26.85, $p < 0.05$, i.e., when move links were chosen, students stated that the movie link was helpful in 74% of the cases.

3. Effects on Language Learning: Thirdly, we assessed the *effectiveness* of the program for two levels of *language learning*: short-term recall of vocabulary items and overall reading comprehension. We first wanted to determine whether there are any effects of the multimedia links on the learning of individual vocabulary words. A vocabulary test of 17 words, as described in Section 2. above, was administered to the students after they had used the program for two 50-minute class periods. Students were not forewarned that they would be tested, so they were not consciously trying to learn new words. The same test was administered after two weeks, again without telling the students that they would be re-tested. Results of both tests, showing the number of correct answers, are presented in Figure 8. Test words are further categorized as to the link which students reported as having helped them learn the word. It should be noted that of the possible number of answers to the vocabulary test (17 words x 44 students = 748), students made only 331 and 293 entries, respectively, on the two tests.

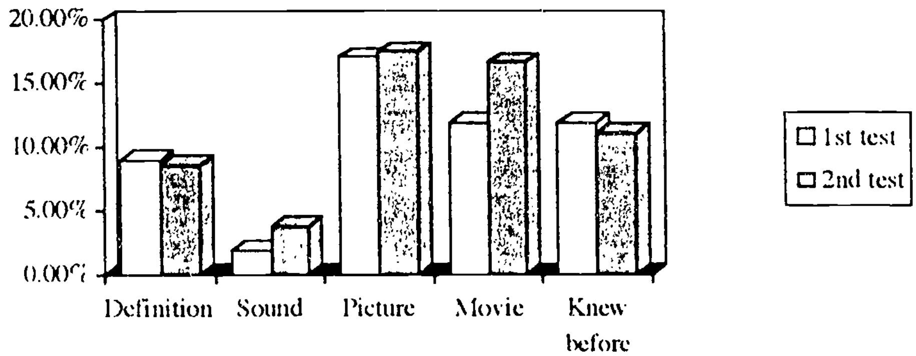


Fig. 8. Correct answers out of total number of entries on vocabulary tests

Statistical tests to determine the correlation between the multimedia links chosen for each word and whether the word was learned correctly did not prove significant (see Table 1). However, a positive relationship was found for all types of links which could be chosen. When text links were chosen for words which had such links available and an answer was given on the first vocabulary test, 77% of the words were learned correctly and only 23% of the answers were incorrect. When picture links were chosen for words which had picture links, 75% of the answers were correct, and similarly, for the words with movie links, 79% were correct.

Table 1. Correlation between media link chosen and vocabulary test results

Vocabulary test result	Text		Picture		Movie	
	chosen	not chosen	chosen	not chosen	chosen	not chosen
total wrong	47	35	13	18	5	16
total right	132	117	60	54	15	60
pct. wrong	14.2%	10.6%	9.0%	12.4%	5.2%	16.7%
pct. right	39.9%	35.4%	41.4%	37.2%	15.6%	62.5%
	N=179	N=152	N=73	N=72	N=20	N=76
	Pearson χ^2 (1) = 0.460, p =0.497		Pearson χ^2 (1) = 1.115, p =0.291		Pearson χ^2 (1) = 0.144, p =0.704	

Striking is the fact that in the second "surprise" vocabulary test two weeks later, almost the same total number of correct answers were given: 249 immediately after using the program for 2 days, and 235 two weeks later. Broken down by type of link, there is in fact a slight improvement in the percentages: of the words with text links, 83% of the answers given were correct; of the words with picture links, 90% were correct; of the words with movie links, 82% were correct.

A significant dependence was found between the self-reported data from the users when the text link was reported as most helpful in learning a word and whether the word was actually correct on the vocabulary test. The Pearson χ^2 (1) = 3.89, $p < 0.05$, i.e., when students reported that a text link had been most useful for remembering a word, their answers on the vocabulary test for that word were correct in 68% of the cases. While no significant dependence was found for the cases where picture or movie links were thought to be most helpful, there was a positive relationship in both cases, and similar percentages of correct answers were given: 75% and 83%, respectively.

In order to test overall reading comprehension, students were asked to write a "recall protocol" of the story. This technique of summarizing a text in one's native language is widely used in assessing reading comprehension in both first and second language acquisition. The hypothesis was that the QuickTime movie preview would provide students with a global meaning of the text and that the other features, such as the movie-text link, which allowed them to hear the entire text read by native speakers, would also contribute to understanding.

However, the pilot test results of the recall protocols showed very poor comprehension. These results can be attributed to two main factors: first, the actual content of the preview, while aesthetically appealing, was much too subtle, and second, there were some technical limitations. These limitations revolved around the fact that the movie (2:45 min.) was too large to load on each student's hard drive and therefore had to be shown to each class as a group at the beginning of the hour. Students were *not* able to watch the movie again, individually, whenever they chose to. In addition, no headphones were available to students, so whenever a student chose a link or played the entire narration of the text, everyone else in the room would hear it. This may have deterred students from using the text-movie link feature, and in fact, only a few ventured to use this feature at all.

The preview is currently being redone and subsequent testing of the assumption that only knowing the meaning of individual words does *not* imply or insure understanding of a text will be conducted. With a more explicit preview, the effects of multimedia information on the higher-level process of comprehension of the text as a whole can then be assessed.

Summary and Conclusions

In summary, in assessing the effectiveness of our multimedia software in improving reading comprehension, we first investigated the usability of the program in terms of clarity of the different links (text, picture, movie). Users found the text links (definitions in German) the least helpful, probably due to the difficulties in understanding the German rather than to the type of link. On the other hand, picture and movie links were ranked as the most helpful types of links. (In the next testing phase, we will both improve the definitions in German as well as add a second "layer" of help, consisting of English descriptions or translations of the words or phrases.)

We then examined the data from the user logs to determine which links were chosen and whether there was a correlation between which links were actually chosen and which were reported to be most helpful. A statistically significant dependence was found for all three types of links. In the case of text links, there was a negative relationship: when a text link was chosen, users did not find this link helpful in more than 50% of the cases. However, in the case of picture and movie links, when either of these links was chosen, users did find the link helpful in 57% and 74% of the cases, respectively.

The data from the two vocabulary tests, administered immediately after the program had been used for two class periods and then again two weeks later, show, first of all, very little attrition of correct answers after the two week interval, and secondly, a positive, although not significant, correlation between the type of link chosen and whether the word was learned correctly. When text, picture or movie links were chosen for words which had such links available, high percentages of correct answers were given (77%, 75%, and 79%, respectively).

In conclusion, students showed a definite preference for the picture and movie links (over the textual and audio links), both in their responses on the questionnaires about which links were most helpful, as well as in terms of which links they actually chose when using the program. While the movies were found to be most useful, the QT movie was, compared to the physical clarity of the textual and picture links, of much lower quality. This suggests that although the quality of the links was uneven across types of links, the more important factor appears to be type of link.

In addition, this is supported by 1) the statistically significant *positive* dependence of chosen picture and movie links with students reporting that these two types of links were helpful, and 2) the statistically significant *negative* dependence when text links were chosen and students reported them as useful.

In terms of the effectiveness of the different types of links for learning vocabulary words, there was a positive, though not significant, relationship between type of link chosen and whether the word was learned correctly. A possible contributing factor which was not included in this analysis is the fact that the program allows multiple links to be chosen in any combination, and therefore attributing learning to a single type of link for each word may not be warranted. Future testing will address this issue.

While this pilot test only suggests that visual types of links or information may be preferable to textual links, future tests will examine the hypothesis that students with different learning styles, e.g. visual, auditory, or textual, prefer different media types of information. If this hypothesis is then confirmed, we could consider this to be strong support for the notion that multimedia systems bring a new dimension to educational software because they can be adapted to individual learning types.

References

Bernhardt, E. B. (1991). *Reading Development in a Second Language: Theoretical, Empirical and Classroom Perspectives*, Norwood, NJ.

Carlson, P.A. (1990). Square Books and Round Books: Cognitive Implications of Hypertext. *Academic Computing 4* (7), 16-31.

Clark, R. E., & Salomon, G. (1983). Media in teaching. In *Handbook of research on teaching*, ed. M. Wittrock, New York, 464-478.

Hulstijn, J. H. (1992). Retention of Inferred and Given Word Meanings: Experiments in Incidental Vocabulary Learning. In *Vocabulary and Applied Linguistics*, ed. P. Arnaud & H. Béjoint, London, 113-125.

Pederson, K. M. (1987). Research on CALL. In *Modern Media in Foreign Language Education: Theory and Implementation*, ed. Wm. Flint Smith, Lincolnwood, IL, 99-131.

Acknowledgements: We would like to thank Stanley Nicholson for his expertise and willingness to help us with the design of the study and statistical analysis of the data and the Office of Instructional Improvement at the University of California, Santa Barbara for a grant to develop *CyberBuch*.

Student Response to Hypermedia In The Lecture Theatre: A Case Study

DAMIAN CONWAY
Victorian Centre for Image Processing and Graphics, Department of Computer Science, Monash University, Clayton, Victoria 3168, Australia
Email: damian@bruce.cs.monash.edu.au

Abstract: The Computer Science Department at Monash University recently began presenting lectures using projection of a hypertext system running on a notebook computer as the primary medium. This paper presents a statistical analysis of student reactions to this approach, focusing on the effects, as perceived by the students, on retention and comprehension of material, lecturer performance and overall benefit of computer-mediated presentation (CMP). Student responses are found to be very positive, with strong correlations between scholastic ability (actual or self-perceived) and student approval of CMP. No significant variations in response are found between male and females, nor between native and non-native English speakers.

A Case Study in Lecture Presentation

Computer-mediated presentation has been discussed for over two decades now (Levien & Mosmann, 1972). Yet, despite the fact that traditional lecture media (that is, blackboards or overhead projectors) impose significant restrictions on the presenter (Penner, 1984; Guskey, 1988; Sharpe & Willshire, 1989), computer-based presentation systems have yet to make significant inroads into the lecture theatre (Dressel & Marcus, 1982).

During the 1993 teaching year the Computer Science Department at Monash University initiated a study to gauge the effectiveness of portable computers as a presentation medium. A hypertextual presentation system, HyperLecture (Conway, 1993a), was developed and used to present an introductory computer programming course at the 1st Year level. Both lecturer and students reacted favourably to the use of the computer as a presentation medium (Conway, 1993b), but no empirical evidence could be gathered until the completion of the course.

During the course review students were asked to complete an additional questionnaire, designed to assess quantitatively their reactions to the use of computer-mediated presentation. The questionnaire asked them to rate the use of the computer in terms of its effect on the lecturer's presentation style, whether it improved information transfer and/or retention, and their overall preference for computer-based or standard presentation techniques

The findings supported the lecturer's opinion that computer mediated presentation had been better received than traditional lecturing techniques. However, some interesting discrepancies in student opinion were noticed when their responses were analysed according to scholastic ability, gender and native language.

The remainder of this section presents a brief overview of the presentation system used. The next section discusses some of the features of the survey and the motivation for particular questions. Sections 3 and 4 present an analysis and interpretation of the results of the study. The final section attempts to draw together the experience of the lecturer and the feedback of the class.

HyperLecture

HyperLecture is an Apple HyperCard stack designed to greatly facilitate the design, creation and presentation of linear and non-linear lecture materials. HyperLecture incorporates many features specifically aimed at minimizing the interference of the presentation medium on the presented message. A full summary of the capacities of HyperLecture is presented in (Conway, 1993a).

Significant features of the HyperLecture stack include:

- A simple gestural command interface, which avoids cluttering the screen with unnecessary controls.
- Optional automatic cross-referencing of materials into a hypertext.
- Multiple navigation modes, supporting linear, hierarchical and free-associative types of presentations.
- The capacity to quickly produce coordinated lecture handouts and summaries, either as electronic documents, reproductions of the actual presentation materials, or as a fully formatted summary document.

- A simple plain-text input format. Input text is automatically formatted into a clear, readable, minimalist style, with optional embedded directives giving finer control over presentation format.
- An auxiliary stack which facilitates the presentation and explanation of programming code examples.
- The capacity to annotate materials textually and verbally for later student revision.
- The capacity to design and implement self-driving audio-visual tutorials using the lecture material.
- Audio assistance for visually impaired users.

Typically, a prepared HyperLecture stack is loaded onto a Macintosh PowerBook and is presented (using the screen mirroring facilities of the PowerBook) via an LCD projection panel on a standard overhead projector. Alternatively, the PowerBook may be plugged directly into a large screen display device.

The Questionnaire

A brief questionnaire was issued to 300 students taking the Introduction to Computing subject. Response to this questionnaire was anonymous but compulsory. The seven questions asked were:

1. Compared to other lecture series you have taken this semester, were the lecture materials presented as clearly using the computer?
2. Did the code demonstrations help you to understand the examples better?
3. Did the code demonstrations help you to remember the concepts presented better?
4. Would you have liked to take an electronic copy of the lecture material home for study and revision?
5. How do you think the use of the computer affected the lecturer's presentation style?
6. Do you prefer computer-based presentations or "traditional" presentations (eg: overhead projector and/or blackboard?)
7. Overall, was the use of a computer to present the course of any benefit in helping you to understand the material presented?

For each question students were asked to select a numerical response from 0 to 10, indicating strongly negative response to strongly positive response respectively. They were also asked to furnish information regarding their previous academic achievement and their expectations. This information was requested to test whether (actual or perceived) academic ability was a factor of a students acceptance of computer mediated presentation. Gender differences were also examined.

Many of the students in the class come from countries where English is not the principle language. Such students often experience difficulty in understanding lecture materials, because of language barriers. Hence, native language (specifically, whether the student was a native speaker of English, the presentation language) was investigated as a third possible factor of student acceptance of CMP.

The Student Response

Figure 1 summarizes the overall student responses to the questionnaire whilst Figure 2 shows a breakdown of responses by gender and native language. Figure 3 shows the breakdown of responses to the two summary questions (questions 6 and 7) according to the actual and predicted scholastic achievement of students.

Observations and Commentary

General Observations

It is clear that neither native langu: e nor gender is a significant factor in the response of the students to computer-mediated presentation of lecture materials. There is a small but consistent difference between the mean response of male and female students across all questions, with males responding 0.5 ratings (5%) more positively than females. This may be a genuine difference in perception or may be due to other factors (for example, gender-specific cultural attitudes towards Computer Science or gender-based attitudes to the presentation style of a male lecturer.)

There was no detectable difference (less than 1%) between the overall response between native English speakers and those whose first language is not English. This may be due to the high standard of English literacy amongst international students (a TOEFL score in excess of 575 is an entry requirement for the course.)

To evaluate the statistical variation of student response with scholastic performance, students were classified according to the letter grade of their results: A (100-85, or High Distinction), B (84-75, or Distinction), C (74-65, or Credit), D (64-50, or Pass), E (49-45, or Terminating Pass) F (44 and below, or Fail).

When marks achieved on a midsemester test were considered, there was found to be a strong correlation (81%) between letter grades and the average response of students achieving that grade. For the two summary questions

Response	1. Were lectures presented as clearly using CMP?	2. Code demonstrations assisted understanding of examples	3. Code demonstrations assisted recall of concepts	4. Would you like an electronic copy of lectures?	5. CMP affected lecturer's presentation	6. Prefer CMP or traditional presentations	7. Overall, CMP was of benefit
0 (very negative)	2.00%	0.33%	0.67%	0.67%	0.33%	1.00%	1.00%
1	2.33%	2.00%	1.67%	0.00%	0.67%	1.67%	0.33%
2	3.67%	2.00%	1.33%	0.33%	0.67%	2.00%	2.33%
3	5.33%	2.67%	3.33%	0.33%	1.67%	2.33%	1.67%
4	7.33%	2.33%	3.67%	0.00%	1.33%	2.00%	4.00%
5 (neutral)	11.00%	7.67%	14.33%	7.33%	8.33%	8.33%	8.33%
6	6.67%	12.33%	16.67%	2.67%	7.67%	3.67%	9.00%
7	17.33%	21.67%	24.67%	7.67%	19.00%	12.67%	19.00%
8	19.00%	16.67%	15.67%	16.33%	26.00%	19.00%	27.00%
9	12.00%	17.33%	11.33%	17.33%	15.33%	12.67%	12.67%
10 (very positive)	13.33%	15.00%	6.67%	47.33%	19.00%	34.67%	14.67%

Figure 1. Summary of student responses

(questions 6 and 7) this correlation was 93%. However, as indicated by the high standard deviation of the responses (2.2 ratings, averaged across all questions), the correlation between the letter grade and responses of individuals show negligible correlation (only 18%). This is reflected in Figure 3.

Analysis of the correlation between students' responses to CMP and their predictions of their final marks show very similar patterns Letter grade and average student response within that grade were correlated at 83% overall and at 89% for the summary questions. Correlation between grade and individual response was, once again, effectively uncorrelated (at 18%).

An interesting exception to the general pattern of high correlation between predicted letter grade and average reaction to CMP is in the response of students predicting an F grade. Averaged across all questions, students who believed they were going to fail exhibited an average response 7% (approximately three-quarters of a rating) higher than students who believed they would achieve a non-continuing pass. This pattern is reflected in the vee-shaped "tails" in the "Predicted final result" graphs of Figure 3.

Three possible explanations for this anomaly present themselves. This exception to the correlation between grade and response may be indicative of wishful thinking on the part of weaker students. Alternatively it may suggest that students predicting failure are an enthusiastic but pessimistic group within the population. Finally it may be that students who expect only a terminating pass have lost interest in the course or confidence in themselves and therefore their reactions to CMP expressing a negative attitude to their studies as a whole, rather than specific hostility to computer-based lecturing.

Nevertheless, it may be concluded that both students who perform better and students who are more confident are generally more positively disposed to CMP, but with large individual variations. Initially it was hypothesised that, given the similarity of statistical and individual response patterns, these two factors – performance and confidence – might be highly correlated. However, analysis reveals a correlation of only 36% within this population, suggesting instead that either ability or confidence suffices as a predictor of enthusiasm for CMP.

Student Response to Specific Questions

68% of students reported that the lecture materials were presented more clearly using the computer; 21% indicated that other lecture presentation methods were more clear. The average rating for presentation clarity was 6.6 out of 10. Males responded 9% more positively than females, with an average rating of 6.8.

Students who predicted a non-continuing pass (E grade) for the subject were the least positive group (the only group with a negative average.) Students predicting an A grade reported greatest positive response – a difference of nearly 4 ratings higher than the students predicting an E. This difference was largest margin between any two groups in any question. Students who expected a failing grade (F) were, on average, 1.5 ratings (16%) more positive than students expecting a non-continuing pass.

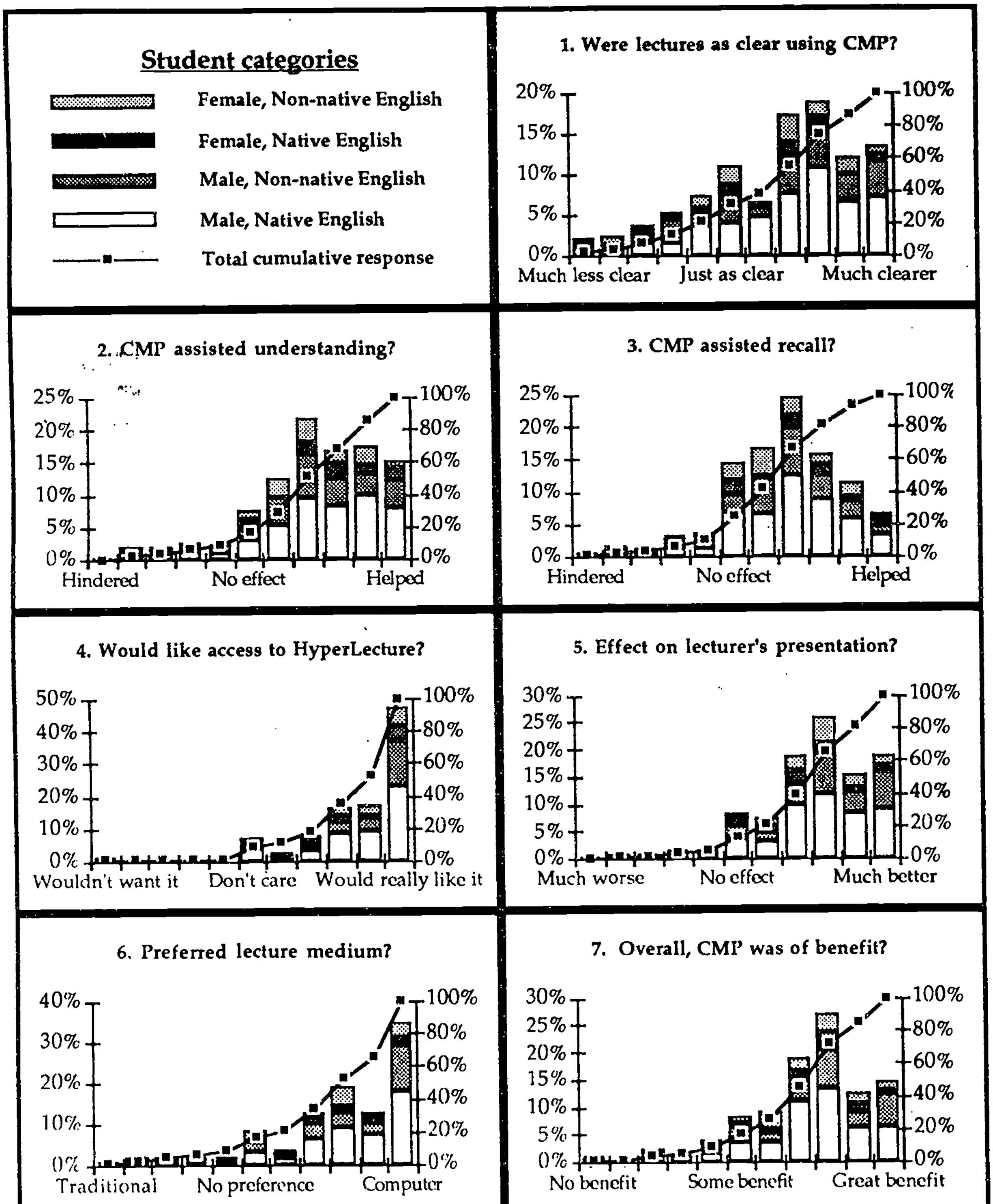


Figure 2. Student responses by gender and language groups

Approximately 83% of students reported that the code demonstration software integrated into HyperLecture system helped them to understand the programming examples to some extent. 9% of students indicated that the software had some negative effect. Average response to this question was a rating of 7.23 out of 10.

As with question 1, students who expected to achieves grades of A or B reported significantly more positive perceptions (approximately 3 ratings) than students expecting a non-continuing pass. Again, students expecting to fail were noticeably more positive about the effects of CMP than those expecting an E.

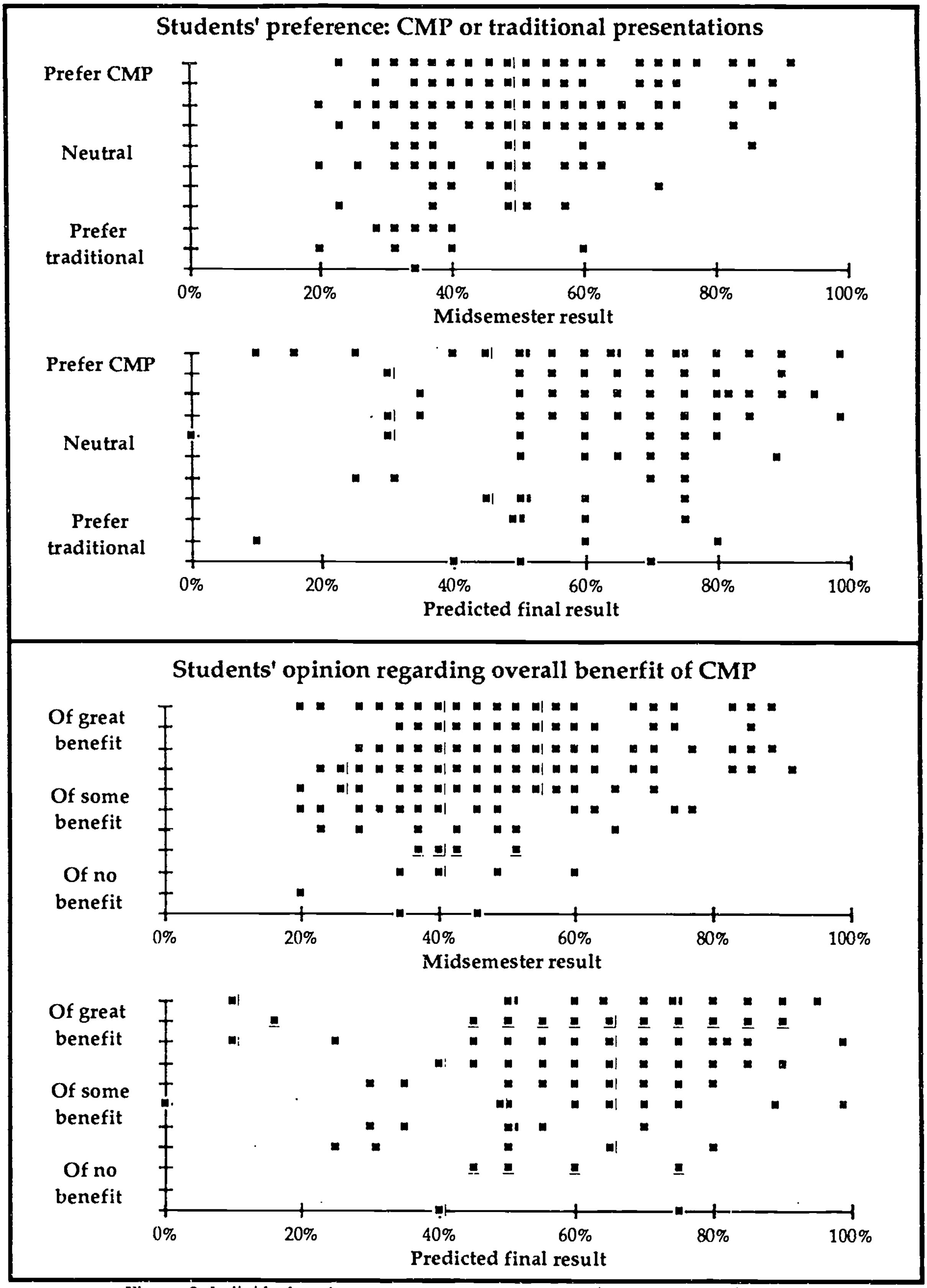


Figure 3. Individual student responses to summary questions, by scholastic factors

Students were less convinced that the use of the code demonstration software aided their long-term recall of programming concepts, with 75% of students reporting some benefit. 11% of students suggested that the software had some negative effect on their recall. The average rating for this question was 6.65 out of 10. Despite the large number of neutral responses to this question, there was still solid correlation (86%) between predicted grade and average degree to which students predicting that grade believed that CMP had assisted their recall of material. However, at 42%, correlation between students' actual grades and their individual response was very poor.

Students exhibited an overwhelmingly positive response when asked if they would have liked to take an electronic copy of the lecture material home for study and revision. 91% of students showed at least some inclination to do so and only 1% showed any disinclination. The average response to this question was 8.70 out of 10, the most positive rating for any question. No doubt this reflects the average student's perennial enthusiasm for any educational resource.

Asked how they thought the use of a computer had affected the lecturer's presentation style, 87% indicated some beneficial effect, with 5% indicating a detrimental effect. The average response was 7.65 out of 10. Both predicted and actual grades were clearly correlated as factors in student response (93% and 83% respectively.)

In response to question 6, 83% of students reported that they preferred computer-based presentations, whilst 9% indicated a preference for "traditional" presentations (for example, overhead projector and/or blackboard.) The average response was 7.82 out of 10. A greater percentage of males indicated preference for CMP, with average male responses being approximately 1 rating level (9%) higher than that of female students. In contrast, there was absolutely no variation by native language group (less than 0.2%).

The final question, which asked if the students believed that the use of CMP was of any benefit in helping them to understand lecture materials, was procedurally flawed in that it failed to allow students to indicate any detrimental effects. Nevertheless 82% of students indicated there at least some benefit, with an average response of 7.30 out of 10. Once again, student response was strongly correlated with both predicted and actual grades.

Conclusion

It is clear that, generally speaking, the students responded very positively to the use of computer-based hypermedia in the lecture theatre. Most particularly, students were extremely keen to interact directly with lecture materials and code examples, by taking copies of them home for their own use. There was a slightly more positive response by male, English-speaking students, but this is probably more a reflection of the gender of the lecturer and cultural attitudes to the lecture subject than suggestive of varying reactions to the technology.

Actual scholastic achievement was well correlated with statistical response to CMP, but showed negligible correlation on the individual level. There was also strong correlation between expected grade and average response for that grade, but again there was wide variation in individual responses.

Students generally reported that the computer-based lecture materials were presented more clearly than materials in standard lectures. It appears that CMP assisted their initial understanding of material more than subsequent recall, although a majority of students reported benefits in both areas. It is possible that this partial failure may have been overcome had students been able to gain direct access to the hypertext materials for revision purposes.

CMP was clearly more popular than standard lecture presentation techniques, although this may partially be indicative of the relative novelty of hypermedia in the lecture theatre. However a clear majority of students also indicated that they believed the use of CMP to have been of benefit to them.

References

Conway, D.M. (1993a), HyperLecture: An Self-correlating Lecture Presentation and Revision System, *Proceedings 1993 East-West International Conference on Human-Computer Interaction*, Moscow, I11-I26.

Conway, D.M. (1993b), Improving Educational Outcomes With Computer-Mediated Presentation, *Proceedings 1993 International Conference on Computers in Education (ICCE'93)*, Taipei, 325-327.

Dressel, P.I. & Marcus, P. (1982), Enriching Learning through Technology, *On Teaching and Learning in College*, Jossey-Bass, San Francisco, 66-84.

Guskey, T.R. (1988), Time Use and Student Involvement, *Improving Student Learning In College Classrooms*, Charles C. Thomas Books, Springfield IL., 83-102.

Levien, R.E. and Mosmann, C. (1972), Instructional Uses of Computers, *The Emerging Technology*, McGraw-Hill, New York, 51-81.

Penner, J.G. (1984), Interest And Attention In The Classroom, *Why Many College Teachers Cannot Lecture*, Charles C. Thomas Books, Springfield IL., 114-133.

Sharpe, D. & Willshire, M.J. (1989), Human Factors In Teaching, *SIGCHI Bulletin*, *20*(3), 58-62.

Learning Network Design: A Methodology for the Construction of Co-operative Distance Learning Environments.

Dick Davies
CECOMM Research Centre, Southampton Institute, UK.
E-Mail: davies@cecomm.co.uk

Abstract: Learning Network Design is a socially oriented methodology for construction of co-operative distance learning environments. The paper advances a social constructivist approach to learning in which learning and teaching are seen as a process of active communication, interpretation and negotiation; offers a view of information technology as a medium of constrained communication; and finally develops an approach to the design of co-operative learning environments.

Interaction over distance and/or time, in its modern sense of rapid exchange, implies the use of telematic media. These telematic media arrive with embedded design attitudes towards their utilisation. The paper will argue that the embedded engineering paradigms that accompany these technologies are inappropriate for the design and construction of interactive social environments and will propose an alternative process-oriented paradigm for the development of a design methodology more appropriate to co-operative distance learning environments.

This paradigm has two strands embedded within it seeing:

a. Learning as Co-ordinated Communication

b. Computer Technologies as Communication Tools

and offers a common **communication-oriented basis** from which to view both learning and information technology and from which to develop a theory-based and **systematic approach to design.**

Approaches to Design

The computing and telecommunications industries developed within a traditional engineering paradigm. This paradigm is linear and uni-directional. Products are specified, designed, manufactured and delivered. Once specified a product is deemed to have entered a production cycle and not subject to change. The focus is on the design of products within technical systems where the formal design methodologies assume that change is costly. These methodologies are constructed to ensure that specification is as complete as possible. Many of the comments by those at the receiving end of this approach show their disenchantment with these formal methods where 'systems designers' produce computing systems that have little relationship to the needs of the end users.

This engineering paradigm although applied to computing and information systems is totally inappropriate and unnecessary as the underlying technologies, unlike physically produced objects, are capable of being restructured on an ongoing basis. In the context of interaction over distance to some extent this move is already evident in the telecommunications sector with the development of INS (Intelligent Network Services) and the related concept of VPNs (Virtual Private Networks) which enable end users to construct, deconstruct and reconstruct multiple logical networks that then overlay actual physical telecommunications infrastructures.

An alternative paradigm is process rather than product oriented and advances that computing and information systems are socio-technical systems and, like 'soft' human systems are, capable of restructuring dynamically. The paradigm is therefore iterative and circular. It recognises that there is a specification or analysis phase and that is followed by design and system development, but argues that non of these phases are closed or linear. In other words the user can, and should be, involved at all stages in the development cycle. In this world there is never any **end product** simply a stage in an ongoing development process.

This is the background to the need to develop a formalised design methodology for the construction of co-operative learning environments. This paper now:

1. Advances a social constructivist approach to learning in which learning and teaching is seen as a process of active communication, interpretation and negotiation.
2. Develops a view of information technology as a medium of constrained communication
3. Finally offers an approach to the design of co-operative learning environments.

1. Learning as Co-ordinated Communication.

'There is no way, none, in which a human being could possibly master the world without the aid and assistance of others for, in fact, the world is others.' (Bruner in Wertsch 1985.) The reason for starting with this obvious statement is that most learning theories ignore this reality. In the many-to-many interactions enabled by the advanced learning technologies based on telematics this social dimension cannot be ignored. An approach to learning is required that can deal with both the social and individual dimensions of learning.

In overview only this section of the paper :

a. Stresses the process of communication, in particular:

i.) the importance of context; ii) the role of 'rules' as 'co-ordination tools'; iii) the purpose of 'role' in the co-ordination of communication;

b. Links the above analysis to communication-oriented approach to learning using the following Vgotskyian-influenced approach:

i. Learning requires, a) the necessary existing cognitive competence, b) the necessary conceptual 'bridge' between the new and the old,

ii. That this bridging process between new and old conceptual structures is an intrinsically social process whether for 'individuals' or for groups

iii. and that, in general, the 'tutor' is a key factor in enabling this 'bridging' or 'scaffolding' between the 'new' and the 'old' to take place to take place in learning.

iv. We therefore need to identify the elements that underlay the communication processes that enable learning

v. and develop an appropriate (collaborative) pedagogy for both individuals and groups in which learning can take place

c. And finally, to operationalise the approach with an emphasis on: i) the role of the tutor in learning; ii) implications for design and for collaboration

In summary, the approach to learning offered takes account of the potential openness in communication offered by the time and/or distant independent technologies of advanced co-operative learning technologies. It offers a socio-cognitive perspective on learning stressing the importance of:

1. Context
2. Co-ordination of action within a contextual framework
3. The active nature of learning
4. The role of the tutor
5. The role of other learners as a resource

(see Davies, 1992 for detailed exposition of above)

2. Computer Technologies as Communications Technologies

This section develops the theme that computer technologies are communication media and focuses on two linked areas:

i. The nature of **interaction** with computer technology

ii. The nature of **integration** of computer technology in learning settings.

In particular it advances that:

1. Computer technologies are interactive communication technologies. They are however:
   a.. technologies of 'constrained communication' (Petri, 1977), and
   b. general purpose non-context specific machines i.e. context-insensitive
2. The co-ordination of action is **medium dependent** and does not automatically transfer to computer-based systems.

From this and in the context of the use of advanced computer technologies for learning collaboratively and otherwise we can immediately identify a list of areas for concern:

a. The effect and role of the media used
b. Nature of the 'constraints' in computer technology
c. Problems of contextualising interaction
d. Co-ordination of interaction in complex learning scenarios via telematic systems.

Petri (1977) sees computers as, *'a general communication medium for strictly organised information flow'* What is the nature of this strictly organised flow? As the surrogate interaction enabled by computers cannot control for external context it must constrain interaction in order to enable dialogue to occur at all. It does this by restricting the nature of allowable dialogue precisely.

The problem with this constraint is that a number of different provider perspectives have to be accommodated to deliver any co-operative learning environment understandable by its users. In particular the important role of metacommunication in learning was outlined earlier. For the co-ordination of action three linked elements were required: a. recognition of context, b. communication rules, c. assumption of role(s). However, computers as general purpose tools for interactive communication do not have a 'context' embedded within them. They are media of constrained communication. They do embody communication rules, but the rules are programmed by their designers. Since the machines themselves are unable to contextualise they require users to **adapt** to their embedded rules. They do this by having the role of 'surrogate partner' embedded within their programming. (Maass 1983). The computers' 'role' as a partner will however be limited to knowledge of the application running and to a generic context. All these points make current computer-based communication system a difficult medium for enabling co-operation.

In summary for a computer-supported co-operative learning environment the following matrix can be developed:

	Co-ordination	**Communication**
Dialogue with Machine	**Machine Controlled**	**Constrained**
Dialogue with People	**Structured by Context and Application**	**Open, but with media constraints to bandwidth**

It now clear computer systems are communication technologies, but with a particular and limited set of interactive capabilities. The role of this section has not been to offer a set of learning methods, but rather to point out that these technologies are in fact communication media. They therefore need to be treated in design terms as any other communication media whilst at the same time recognising their important interactive capabilities and disadvantages. In this sense they are enabling in that they can act as a communication medium for co-ordinating interaction across time and distance. (Davies, 1992 for further details)

3. Learning Network Design

Given the socio-cultural approach to learning with advanced interactive communication technologies advanced here - *Computer-Supported Co-operative Learning* (Davies, 1988) - what does this imply for the design of distance learning environments in the light of the comments above? O 'Malley(1990) sees two main issues for the design of collaborative systems:

1. 'how to construct an environment in which learners can share representations and contribute equally to the joint activity'
2. 'how to design the appropriate representations and tools which will make it possible for learning to take place'
The focus is therefore on the design of collaborative communication media and tools for both its initial construction and end-user manipulation. This section offers a process approach to design and develops a methodology for the production of co-operative learning environments

3.1 Designing With the User

The 'user' in technology-based learning systems is a problematic concept. Is it the learner, the tutor, the author, the course manager, the administrator, ...? In large scale telematics-based environments it is all these users. The point is that each of these 'users' has different perspectives on what is essentially an 'organisation'. This shared 'environment' can be sliced into a number of logical views depending on the viewpoint of the 'user'. It is therefore the role of the service provider and of Learning Network Design to accommodate these views by providing the design tools to enable their conceptualisation and ultimately their construction.
Learning Network Design is a design methodology developed from the 'soft systems' design methodology of Checkland and suitable for the design of 'loosely coupled systems'. Typically, these systems are human organisations: *'By 'methodology' I do not mean method....My sense of the word here is that the outcome of the research is not a method but a set of principles of method which in any particular situation have to be reduced to a method uniquely suitable to that particular situation.' Checkland (1979.)* Learning Network Design methodology, when applied in a particular context, will give rise to a localised design method appropriate to that context. LND therefore, is not simply about the design of the technical components of learning systems it also concerns the total design of the learning environments AND the parameters to action within them.
The parameters to Learning Network Design are:

1. Methodology as Generic:
Problems are often generic, but solutions are generally local. LND therefore, is a methodology rather than a method. It may spawn several actual methods that are 'situated' in the cultural sense (both in the wider national or regional sense and in a local sense in terms of say, an organisation)

2. Focused on Providing Solutions for Learning in Co-operative Situations
To enable the process of learning in groups

3. Scaleable
Can handle a pair (tutor and student) or the organisational complexity of a major training provider with many thousands of students

4. Simple & Structured
Consists of few elements and so able to be employed with course providers

Learning Network Design therefore is concerned with the design of **'virtual learning environments'**. That is, of enabling the creation of the integrated communication and co-ordination structures and tools that will assist the learning process in time and/or distance learning environments. These are:

a. the system/infrastructural architecture ('context and communication structures')
b. tools for action inside the environment ('communication resources')

and project the views in this paper on computer technologies and the nature of learning.

Learning Network Design is a process rather than product approach to design with the following parameters:

'- relevance with respect to setting
-"quality" defined by looking from the user to the program - to assess programs 'relevance' or 'adequacy' with respect to a particular form of work

-people's competence defined in relation to their own professional practice - and their competence to be continually changing as part of a learning process
-'referent system' composed of "human work, learning and communication" - so in process of continual change and modification .' Suchman (1988)

These issues are addressed within the multiple user perspectives enabled by Learning Network Design with the focus being on the design of communication structures to aid teaching and learning processes. In particular, the fostering of the social processes that will enable cognitive development.

3.2 Learning Network Design

The focus of LND is on the importance of communication in context. LND does not specify an exact specification, rather it provides the parameters for design 'with the user'. These parameters point to the importance of communication and the co-ordination of action at the levels of :

- **Context**
- **Group**
- **Individual**

and focused on :

- **communication**
- **co-ordination of action**
- **role.**

This gives the following generic grid for Learning Network Design:

	Context	**Group**	**Individual**
Communication	Immanent	Structured	'Tool' Use within medium
Co-ordination	Via embedded communication structures	Via dynamic structuring	Mental Models
Role	Embedded performatives	Logical Attribute	'Plans for Action'

This above matrix is the generic design framework for CSCL systems. Design methods should evolve in context from this framework. It can now be operationalised in the task oriented context of particular course requirements. For example it has been used as the basis for the construction of the **'Multimedia Teleschool'** (Berlitz, 1993). This gives the following outline results for the design of a distance language course for delivery via CMCS:

	Context	**Group**	**Individual**
Communication	Constructed metaphor - *'Virtual Language School'*	Closed Task-oriented groups	Written world
Co-ordination	CMCS Embedded Structures - 'Seminar Rooms'	Tutor Directed Learning Structures	Interaction in Learning Group
Roles	Tutor Learner Administrator	From Context and Group Maintenance	Dynamic role 'playout'

Further detailed design will emanate from this matrix at the level of sub-context, sub-group and individual.

4.Conclusion

The approach offered in this paper has been to:

1. Offer a social constructivist pedagogy focused on the role of communication and its co-ordination
2. Argue that computer technologies are media of communication.
3. Develop a linked communication-oriented design methodology - Learning Network Design

In summary, Learning Network Design is a structure for design. As with for example, the design of any organisation, the actual design process should be implemented by those who commission (the users) and by those who advise and implement. The actual implementation is an iterative process of proposal, criticism and reproposal. Learning Network Design offers a framework for design for the realisation of advanced co-operative distance learning environments.

References:

Berlitz (1992) 'The Multimedia Teleschool - Functionalities and Pedagogic Scenarios' European Commission DGXIII DELTA Project D2021 Deliverable 3.1. The Multimedia Teleschool. *Berlitz European Projects*, Frankfurt.

Berlitz (1993) The Multimedia Teleschool. *Berlitz European Projects*, Frankfurt.

Bruner J. (1985) Vygotsky: a historical and conceptual perspective. In *Culture, Communication and Cognition: Vygotskian Perspectives. J.W. Wertsch (ed) Cambridge University Press, Cambridge.*

Bruner. J (1986) 'Actual Minds, Possible Worlds' *Harvard University Press*

Checkland P.B. (1979) 'Techniques in 'Soft' Systems Practice Part 1: Systems Diagrams - Some Tentative Guidelines' *J.App.Systems Analysis, 6, 33-49.*

Davies D (1988) ' Computer-supported Co-operative Learning Systems' *PLET 25 (2)*

Maass S. (1983) Why Systems Transparency?. In Green T, Payne S.J, & Van der Veer G.C. (eds) *'The Psychology of Computer Use'* Academic Press, London

O'Malley C & Scanlon E. (1990) 'Computer-support Collaborative Learning: Problem Solving and Distance Education' in Kibby M. 'Computer Assisted Learning' *Proc. CAL'89 Conference.* University of Surrey, UK. 11-14th April, 1989.

Petri C.A. (1977) 'Communications Disciplines' in Shaw B. (ed) 'Computing System *Design' Proc. J. IBM. Seminar* at University of Newcastle-on-Tyne, UK. 7 - 10 Sept. 1976.

Suchman L.(1988) Review of "Computers and Democracy: A Scandinavian Challenge' (1987) by Bjerknes, Ehn P, Kyng M (eds) Gower Press, Brookfield, VT. In *ACM Trans. on Office Info. Sys.* 6 (2), 173-183

Towards an analysis of visual media in learning: A study in improving syllogistic reasoning.

M.W.DOBSON
Institute of Educational Technology
The British Open University, Walton Hall, Milton Keynes, MK7 6AA
E-mail : m.w.dobson@uk.ac.open

Abstract: People find difficulty in performing syllogistic reasoning. This paper outlines some reasons for poor syllogistic reasoning. Representational systems (RSs) used in presenting and evaluating syllogisms can be distinguished in terms of attributes such as specificity, expressivity and abstraction. Graphical RSs have pedagogical potential because of their expressivity and specificity attributes. The paper looks at different graphical RSs for supporting syllogistic reasoning and describes them in terms of those distinguishing attributes. Secondly it makes claims about which should generate better learning outcomes as a result of the different degrees of these attributes. The talk will report findings of an empirical study relating outcomes from the use of these systems to the systems attributes.

1. Syllogistic Reasoning

Syllogistic reasoning, is the purest form of deduction, has been the study of cognitive scientists and philosophers for more than two thousand years since Aristotle [Aris36] and turns out, in certain cases, to be very difficult for human problem solvers to perform. A syllogism is made up of two premises that each contain a single quantifier, and a conlusion which relates the two end terms;

Some philosophers are polymaths
All polymaths are intelligent
Therefore, some philosophers are intelligent

Premises and conclusions of syllogisms can be in one of four distinct moods:

All A are B	affirmative universal	(A)
Some A are B	affirmative existential	(I)
No A are B	negative universal	(E)
Some A are not B	negative existential	(O)

There are four possible figures, which the premises of a syllogism can be written in. In each case, A and C are the end terms, (occurring in conclusions), and B is the middle term (occurring in both the premises). These figures are as follows;

A - B	B - A	A - B	B - A
B - C	C - B	C - B	B - C

The first two are asymmetrical because the middle term is located in different places in the two premises, and the second two are symmetrical because the middle term is located in the same position in both premises. Each premise can be in one of the four moods (A,E,I,O), and therefore there are 64 distinct forms in the premises. In fact according to more modern views of syllogistic reasoning there are 512 possible syllogisms, because, each pair of premises can be combined with eight possible conclusions, (four conclusions in the form A - C, and four of order C - A). This however does not alter the fact that only 27 of these pairs are valid. Some combinations are very simple and others, people get no better than chance results of success.

1.1. Aspects of Human Performance in Syllogistic Reasoning

People find syllogistic reasoning hard, what is still a mystery is why people find some syllogisms so hard. There are a number of hypotheses to suggest an answer to this question and they fall under the broad categories outlined below.

> "A theory of syllogistic performance should at the very least account for the relative difficulty of different forms of syllogism for the figural response bias, and for the nature of the erroneous responses, including those of the type, "no valid conclusion" (interrelating the end terms). The atmosphere effect can only account for some errors and not for those of the form "no valid conclusion". It is not intended to deal with the relative difficulty or with the figural effect. The conversion theories certainly account for some errors and for some aspects of the relative difficulty of syllogisms, but they cannot explain either the figural bias or the erroneous "no valid conclusion" responses. The Euler circle theories explain some aspects of relative difficulty, but not the figural bias" [Johns92].

The following sections show an overview of relevant theoretical positions with respect to these difficulties in syllogistic reasoning, it is not intended to be an exhaustive summary of the area, but an introduction to current and recent thinking in the area.

The Atmosphere Hypothesis: Woodworth and Sells [Wood35], connoted the term "atmosphere effect" which predicts the errors in performing syllogistic reasoning to be the result of global impression produced by the premises rather than on the basis of strict logical deduction. Atmosphere is defined in terms of two dimensions, quality and quantity. The quality refers to the premise being either affirmative or negative, the quantity of the premise refers to whether the statement is universal or particular. The orthogonal pairings of values subsumed under quality and quantity yield the four moods above (A, I, E, O). The atmosphere effect can then be stated, firstly referring to quality, that: "whenever the quality of at least one premise is negative, the quality of the most frequently accepted conclusion will be negative, when neither premise is negative, the conclusion will be affirmative" and the second principle, referring to quantity, states that; "whenever the quality of at least one premise is particular, the quantity of the most frequently accepted response will be particular; when neither premise is particular, the conclusion will be universal"

Illogical Conversion: Chapman and Chapman [Chap59], have a view on the difficulty of syllogistic reasoning which they called illogical conversion, this is discussed in [News90]. The principle refers to the acceptance of the converse of either an A or an O statement when logically this is not permissible, and where the premise pair has a logically valid conclusion in another figure. This principle accounts for the main errors in AA, AE, AI, IA, AO and OA pairs.

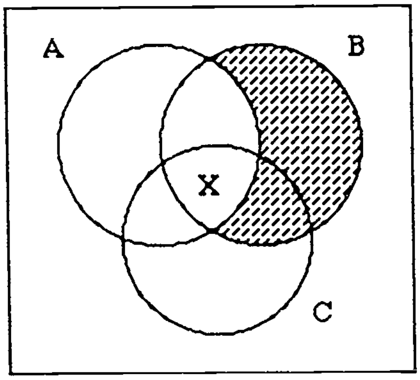
The effects of belief on syllogistic reasoning: Possibly the most pervasive, and in some ways easily countered group of problems with reasoning go under the heading of belief related. There are three principle ways in which beliefs could affect reasoning: (1) They may distort the interpretation of the premises.(2) They may influence the deductive process, biasing which conclusions are reached. And (3) They could be used to filter out unacceptable conclusions that are produced by the deductive process.

2. Representation of Syllogistic Reasoning

There are a number of representational systems which can be used for the support of learning and solving syllogistic reasoning problems. The larger context of this work has involved analysis of many different systems, inluding Venn [Venn94], Johnson Lairds mental models [John92], Eulers circles [Eule72], Stennings adapted Euler cirles [Sten92(a)], Tarskis World [Barw90] and Lewis Carrolls "Game of Logic" [Dodg96]. There is space here only to describe four of these systems.

2.1. Venn Diagrams

Venn diagrams can be used to represent three term quantified deduction (syllogistic reasoning). Three overlapping circles are placed within a square which counts as the universe of discourse. Each of the circles represents one of the terms used in the syllogism. The Venn diagram is used during the process of validating the syllogism by taking each premise and shading out the area where there is no possibility of there being a member within that region, and by marking with a cross those areas where it is claimed that a member exists. If a premise states that a member exists within an ambiguous area, i.e. one where there is two possible areas, then the cross is placed on the line between the two areas.



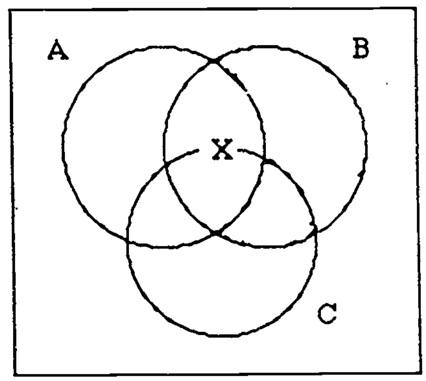


Figure 1 (a) . Showing Venn diagram representation of the syllogism All B are A, Some B are C. Figure 1(b). Showing the partially specified individual Some As are Bs and either C or Not C.

In *Figure 1.* the syllogism 'All B are A, Some B are C' is shown using the Venn RS. The Venn RS is capable of substantially more abstraction than Euler Circles. Venn's system is usually used with a notation for expressing disjunctions (either a chain of `x's or placing symbols on borders). This means it can express partially specified individuals (things which are A and B and *either* C or not C), and so clearly has less `specificity' [Sten93]. Figure 2 demonstrates exactly this partially specified individual using the second of the conventions for expressing disjunctions.

2.2. Euler Circles

Euler circles can be used to represent syllogisms. However there are at least two variants on the Euler circle system. The more traditional system, (see *e.g.* [Cera71]), is described here. In the next section, a more novel interpretation of Euler due to [Sten92] is given. In this traditional system the initial premises are given as the relation between two circles either separated, overlapping one contained with the other, the second contained within the first, or one identified with the other. These relations are known as the Gergonne relations. One of the interesting results of Euler circle RS is that they can give some indication as to the complexity of the syllogism [Cera71], that is, the possible A - C, C - A combinations. It is possible that people when presented with written or spoken syllogisms, believe there is only one diagrammatic representation and therefore one model which relates the premises. If then , when they have constructed a diagram they are unable to immediately read off the conclusion, subjects will have no reason to construct further diagrams. On this theory, subjects will fail when there is more than one Euler representation of the premises and where a conclusion derivable from the first diagram was not derivable from further diagrams which where not constructed.

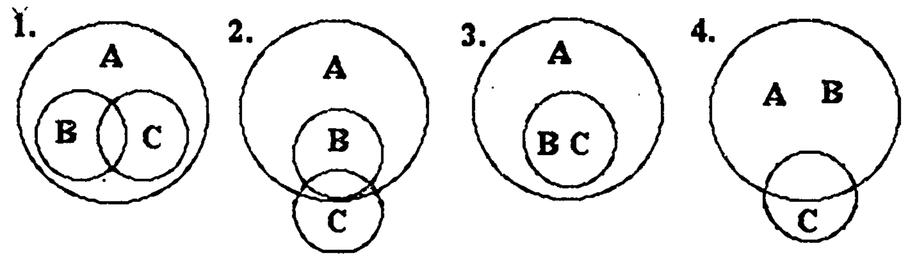


Figure 2. Possible A-C Relations for the syllogism All B are A, Some B are C.

In Figure 2 we see the possible Euler representations for the syllogism - All B are A, Some B are C. The reader will note that in (2), the universal quantification is restricted to the B - A relation. (2) is also the diagram with the "maximal number of regions". With the added convention of some shading in the region where there is certainly some membership, this diagram approaches the adapted system of the next section.

2.3. Stennings Euler circles

Although Stenning [Sten93c] insists his system is a more reasonable interpretation of the Euler system than *e.g.* [Cera71], and indeed the author is inclined to agree, it is useful to separate the two. Conventionally, the four possible premises (AIEO), could be represented by 5 possible circle configurations, (A by 2, I by 4, E by 3 and O by 1). Stennings system with its added conventions for interpretation has an exactly one to one mapping for the premise configurations to diagrams. There are therefore four diagrams and four premises where one premise is represented by one diagram only. Figure 5. shows the Stenning representation of the syllogism All B are A. Some B are C, Some C are A.

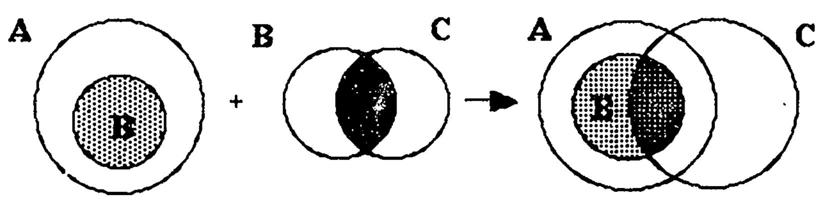


Figure 3. Adapted Euler representation of the syllogism All B are A, Some B are C.

The important features of Stennings system are in the representation of each of the premises and the systematic combination of the premise pairs. So in terms of the premise representation, looking at the second premise in figure 3 (Some B are C), where in the Gergonne relations there would have been four possible diagrams, there is now only one. The added convention of the shading, indicating the existence of a member of that region, now allows for a much more simple representation. The difference can also be demonstrated by a linguistic comparison. Whereas the traditional Euler circles may have allowed for Some As are Some Bs, Some and indeed All As are All Bs, Some As are All Bs and lastly Some but only Some As are Some and only some Bs, the Stenning model permits only the last of these. In terms therefore of specificity, Stennings system is far more specific, less abstract and equally expressive than its traditional counterpart. The traditional Euler system required that the solver created all possible combinations of all possible premise representations, and then to solve the syllogism the solver would have to propose a solution and check that it was consistent with all the resulting diagrams. The enormity of this task can be understood, when, in the case of the syllogism "Some A are B, Some B are C" 16 conclusion diagrams are generated. It is suspected that, although the Stenning systems is a much more simple system to understand and read conclusions from, the process of registration of the two premise diagrams is not so well adopted by the learner. This process is developed more fully in

[Sten92]. Rather simplistically, the process begins with development of characteristic diagrams of each premise, these are given. Registration is a more complex process, the goal being to combine the middle terms circle with as many regions as possible relating the end terms. When this is complete it remains to observe the shaded areas from the component premises , if any of these remain non-intersected, and no more than one premise was negative then there is a valid conclusion. The formulation of the conclusion depends on the quantification of this non intersected area.

2.4. Tarski's World (TW)

Written by Jon Barwise and Jon Etchemendy of Indianna and Stanford Universities respectively, TW is an interactive program designed to be used with the book "The language of First Order Logic" [Barw90]. The program is described by [Gold92], and basically supports an interpreted first order logic for a first order language with equality. The program has three components, a world module, a sentence module and a keyboard module, each in separate windows. The world module displays certain objects, (cubes, tetrahedra and dodecahedra) and certain relationships e.g. Large(x), Larger(a,b) and Leftof(c,d). Although the system is not designed either to teach or support syllogistic reasoning, it is possible to use TW to display states of a world and look at syllogistic inference.

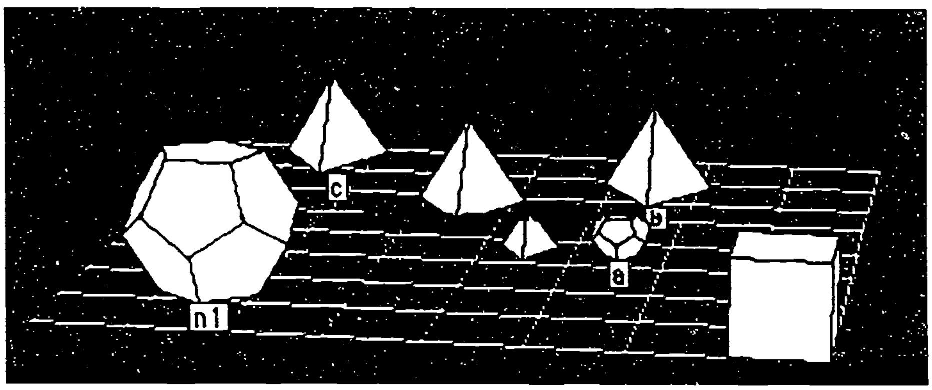


Figure 4. Showing Tarski's World Representation of blocks world.

Materials have been developed which are to be used in association with TW, that help learners to see the relationship between the diagram and a syllogism. The important ideas which can be learned from TW are those associated with the contingency of truth upon the world. In TW where the world is very restricted this kind of message is quite easily presented. When a pair of premises are presented to a student, he may attempt to construct a conclusion. The relationship between the premise and the world is such that there are many worlds in which the premise may be true. If we say , for example, that $\exists x(Large(x) \wedge Tet(x))$, there may be one, two, three or however many large tetrahedra, there may be many cubes, no cubes, dodecahedron etc., in order for the premise to be true. The validity of a conclusion rests on it being impossible to generate a diagram in which the premises are true and the conclusion false. TW fills a further role, which is to show in a step-wise fashion, by the playing of the Hintika game, the validity of any claim a student makes about the truth of a sentence.

3. Empirical Work

Empirical work is under way which will involve the construction of similar tasks which can all be performed using the systems outlined. The method will be to pre and posttest subjects with premises, and conclusions from ten selected prototypical syllogisms with non-lingiustically biased predicates and to ask subjects to validate conclusions. The test population will be sixth form 18 year old school pupils. The results will refer the learned outcomes to the attribute profiles of each of the representational. systems as shown in table 1. (below). Item analysis will reveal the importance of each RS, and attributes of each system in terms the of the various groups of biases from the literature.

	TW	Euler	StennEuler	Venn
Expressivity	High	Low	Lowest	Low
Specificity	High/Low	Med	Low	Med
Naturalness	Med/Low	Med	Med	Med
Abstraction	High/Low	Med	Med	Med

Table 1. Showing qualities of representational systems for solving Syllogistic problems

The results will provide a profile of the most critical factors of the RSs for learning . It is likely that this attribute profile will be transferable to more ill-formed domains, and this will be the subject of later follow-up studies.

5. Acknowledgments

I should like to thank Tim O'Shea (British Open Univerisity) for his invaluable help in the development of the ideas and methods included in this paper, Keith Stenning (HCRC, University of Edinburgh) for comments on drafts and Paul Lefrere (British Open Univerisity) for his encouragement. This work was enabled with the assistance of an Open University Research Award.

References

[Aris36] Aristotle, (336 BCE), Prior Analytics. Tredennik H. (Trans). Harvard University Press, Cambridge Massachusetts 1983.

[Barw90] Barwise, Jon and Etchemendy, John. "Visual Information and Valid Reasoning" in Visualization in Mathematics, Zimmerman and Cunningham (eds.), Mathematical Association of America, 1990.

[Cera71] Ceraso, J. and Provitera, A. (1971) Sources of error in syllogistic reasoning. Cognitive Psychology, 2, 400-410.

[Chap59] Chapman, L.J. and Chapman, A.P., Atmosphere effect re-examined Journal of Experimental Psychology. 8, 220-226 (6)

[Dodg96] Dodgeson, C. Symbolic Logic. Chapter III in Lewis Carroll's Symbolic Logic: ed W.W. Bartley. Hassocks, Sussex: Harvester Press.

[Eule72] Euler, L. (1772) Lettres a une princesse d'Allemange, Vol 2. Sur divers sujets de physique et de philosophie. Letters 102-108.

[Gold92] Doug Goldstone and Steve Reeves "Using programs to teach logic to computer scientists". Department of Computer Science Queen Mary and Westfield College University of London.

[John92] Johnson-Laird, Phillip N. and Byrne, Ruth M.J. Deduction (Precis.) Hillsdale, NJ: Lawrence Erlebaum and Hove, Sussex, UK (1991).

[News90] Newstead, S.E. (1990) Conversion in Syllogistic Reasoning. In K. Gilhooly, M.T.G Keane R.Logie and G.Erdos(Eds), Lines of thought : Reflections on the psychology of thinking (Vol. One). London. Wiley.

[Sten89] Stenning, Keith., Modelling memory for models. In Esquuerro, J.(ed) International Colloquium of Cognitive Science, University of San Sebastian, San Sebastian, 1989.

[Sten92(a)] Stenning, Keith. and Oberlander, J. A Cognitive Theory of Graphical and Linguistic Reasoning; Logic and Implementation. Research paper HCRC/RP-20 May92. University of Edinburgh.1992

[Sten92(b)] Stenning, Keith. Conceptual and Empirical Issues. In Rogers, Y., Models in the Mind: Theory perspective and applications.

[Sten92(c)] Stenning, Keith., and Oberlander, Jon., Implementing Logics in Diagrams . Draft HCRC Report University of Edinburgh Jan24. 1992.
[Sten93] Stenning, Keith., personal communication.
[Venn94] Venn, J. (1894) Symbolic Logic. London: Macmillan
[Wood35] Woodworth, R.S. and Sells, S.B. (1935) An Atmosphere effect in formal syllogistic reasoning. Journal of Experimental Psychology. 18, 451-460.

Multimedia-Supported Intelligent Computer Assisted Instruction: a Spatial Journey into the Brain

Yehudit J. Dori
Department of Education
in Technology and Science

Dov Dori
Faculty of Industrial
Engineering and Management

Technion, Israel Institute of Technology
Haifa 32000, Israel

Jerome M. Yochim
Department of Physiology and Cell Biology
University of Kansas
Lawrence, Kansas 66045, USA

Abstract: Understanding science and technology issues requires primarily a number of spatial abilities to comprehend the 3-dimensional structure of systems. Adequate spatial abilities have long been identified as a necessary mental tool for understanding science and technology phenomena. Multimedia technology has opened the door for a large variety of options that significantly enhance Intelligent Computer Aided Instruction (ICAI) in general and students' spatial aptitude in particular. We have been utilizing a number of multimedia-supported visualization techniques in a studyware of human physiology. This paper presents these techniques using the Brain module as a case in point. The techniques include looking inside the brain, involving the student in exposing brain parts, viewing the brain from different aspects and linking structure with function.

Theoretical foundations of the teaching/learning process on one hand and recent advances in multimedia technology on the other hand, have made possible the creation of a learning environment that benefits from both. This is the basis for the analysis of an hyper-environment – shell – for Intelligent Computer Assisted Instruction (ICAI) development. Within this ICAI shell, science and technology studyware can be developed most efficiently, using state-of-the-art technology and science education methods (Dori, Dori & Yochim, 1992). The division of the learning environment and science or technology domain of study, into two components of a studyware opens the way for the development of a generic framework that supports studyware authoring. Such a framework, or "shell," should enable the expert team, charged with the curriculum development, to concentrate on a sound presentation and teaching of the subject matter. At the same time, the shell is supposed to take care of putting the subject matter in the right educational context, relieving the developers from the burden of low level programming, such as graphic user interface and repetitive tasks.

We have adopted this approach in the development of the *Human Physiology* studyware. The idea of standardizing elements in the software under development, which eventually has turned into a framework for a shell definition, has evolved as the individual modules were being developed. The first module was *The Endocrine System* (Yochim & Dori 1993A). It already contained many of the principal elements that were later reused and improved in the subsequent modules:*The Nervous System* (Yochim & Dori 1993B) and *The Digestive System* (Yochim & Dori 1993C), but several important features were added. These include incorporating test barriers consisting of randomly selected sets of questions, as well as navigation and orientation aids.

Multimedia as a Means of Improving Spatial Ability

Multimedia is a key component of intelligent computer assisted instruction. It is a superb tool that enables the ICAI module domain expert developer to express and convey his/her knowledge about systems and their behavior. Grasping how systems' components interact with each other in both the spatial and temporal domains is the key to understanding science and technology issues.

Spatial understanding of complex structures, as the ones that occur in biological systems, is an important cognitive ability, which involves 3-dimensional thinking. According to McGee (1979), spatial ability is the ability to mentally juxtapose, orient, manipulate, rotate, twist and invert pictorially presented objects. Linn & Peterson (1985) subdivide spatial abilities into three categories: (1) Spatial perception: the ability to determine spatial relationship with respect to the orientation of one's own body. (2) Mental rotation: the ability to mentally rotate a two- or three-dimensional figure rapidly and accurately. (3) Spatial visualization: a problem solving ability of tasks involving multi-stage manipulations of spatially presented information. A significant number of studies (e.g., Small & Morton, 1983; Lord, 1985; and Ben Haim, Lappan & Houang, 1988) have shown that spatial skills can be improved through learning experiences. Linn & Peterson (1985) and Lord (1987) have found gender differences in spatial abilities which can be improved by meaningful interventions. Differences in spatial abilities have also been found between science majors and non-science, liberal arts counterparts, favoring science majors.

These findings point to the significance of spatial aptitude to all science disciplines. Science educators should therefore construct and reinforce spatial thinking and mental formation and manipulations of images in their students. This way they can reach their true potentials and be more scientifically oriented. Multimedia is an ideal tool for materializing these recommendations in science in general and for human physiology systems in particular. The integration of text with graphics has created a host of new opportunities of explaining and demonstrating complex systems' structures and their relation to functions.

Looking Inside the Brain

We have taken advantage of the great variety of options offered by multimedia in the *Human Physiology* studyware discussed above. It comprises a series of modules for teaching and exploring different topics in human physiology. All the modules were developed using HyperCard on Macintosh computers. Each module can be used separately or in conjunction with the other modules, thus can be considered a separate volume – electronic book – in an electronic encyclopedia on human physiology. In this paper, we elected to concentrate on the Brain module as a case in point. From the opening screen it is possible to access directly different subjects included in the module without the need to go sequentially through them, thereby enabling quick access to any desired topic, assisted by on-line help. An even more elaborate access and navigation mechanism is provided by the map of Learning Path, shown in Figure 1. It is a tree structure of the three different levels of detail depth, and cognitive aptitude that the studyware encompasses.

The first level of the Brain module has three topics: Introduction, Sensory Input and Motor Output, and Processing the Signals. Each topic is accessible by clicking the mouse at the name of the topic. Each of the sub-topics in the second level of complexity is recorded next to a "radio button" pointed to by a gray arrow, such as The Somatic Senses in topic B. The sub-topics, too, are accessible by clicking at their buttons. Finally, the most detailed information appear in the subsections, which are the "leaves" of the subject tree hierarchy, and they are directly accessible as well. Taste, Vision, and Audition are examples of such "leaves" in the Somatic Senses sub-topic.

The problem of "getting lost" in the web of possible paths of complex multimedia documents of the order of magnitude of *Human Physiology* is well known to both users and developers of such environments. The Learning Path (LP) map is an invaluable tool for both orientation and navigation throughout the study session. Each card (screen) in the module is equipped with the button "Where Am I?," which can be seen in each one of the following screen images. This button, when clicked, invokes this map and causes the appropriate topic, from which the student asked where he was, to blink repeatedly, until the

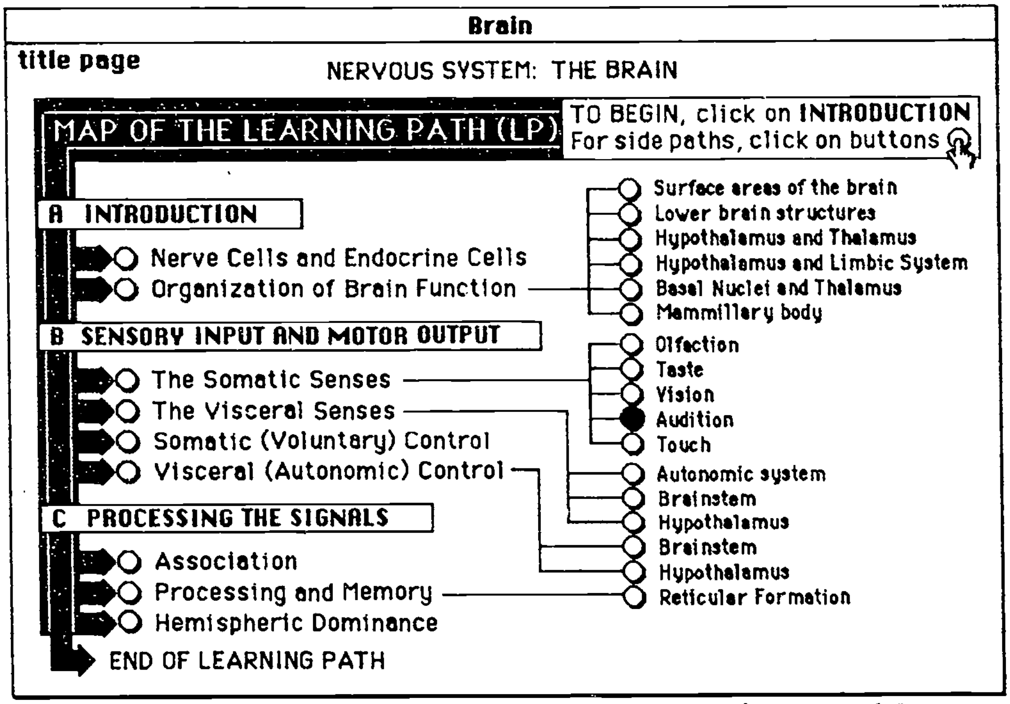


Figure 1. *The Learning Path map of the Brain studyware module*

screen is clicked again somewhere. From the map, the student can either return to where he came from or proceed to another topic. In Figure 1, for example, the student has come from the Audition leaf, which, in the actual study unit, is blinking.

Fostering Mastery Learning Through Exposure of Brain Parts

Following a number of introductory cards, the student is already asked to actively participate in the process of exposing the brain parts. Each boldface word in the text of Figure 2, (such as **brainstem**, **hypothalamus**, and **limbic system**) when pointed to by the mouse (even without clicking) causes the exact location of the corresponding brain part to be marked on the brain drawn to the right. Simulating a brain dissection, which in reality is done only by medical students, if it is at all feasible, with multimedia becomes an easy task that can be performed repeatedly back and forth with no extra cost or effort. By clicking at the left hemisphere of the brain, as the student is instructed to do in another screen, the left hemisphere is removed, and using a visual effect built into HyperCard, the picture is dissolved into a new screen, exposing the brain's inner components.

Having exposed the inner brain's parts, by moving the cursor around the various parts, the name of each part shows up as the cursor enters its area and disappears as the cursor exits it. This is implemented by setting buttons or fields that respond to an "`on mouseEnter`" HyperTalk command. By providing this option, the student has as much time as he or she needs to explore the newly exposed parts. Moreover, by clicking on the back radio button, it is possible to return back to the first card and repeat the simulated hemisphere removal over and over again. This pattern of providing for the iteration of almost any significant visual operation, is used throughout the studyware. This mastery learning principle enables the student to iterate any dissection, rotation, or animation sequence, until he or she is satisfied with the level of spatial understanding of the 3-D structure of the parts being displayed and manipulated.

Animation: Viewing the Brain from Different Aspects

A student electing to deepen his spatial abilities can do so by viewing the brain structure by a series of moving pictures. The pictures appearing one after the other are arranged as a movie, creating the illusion of a rotating brain that is viewed by the spectator from all the important angles. Four of the 12 pictures of this brain movie are presented in Figure 3. This movie goes on repeatedly until the student deliberately stops it.

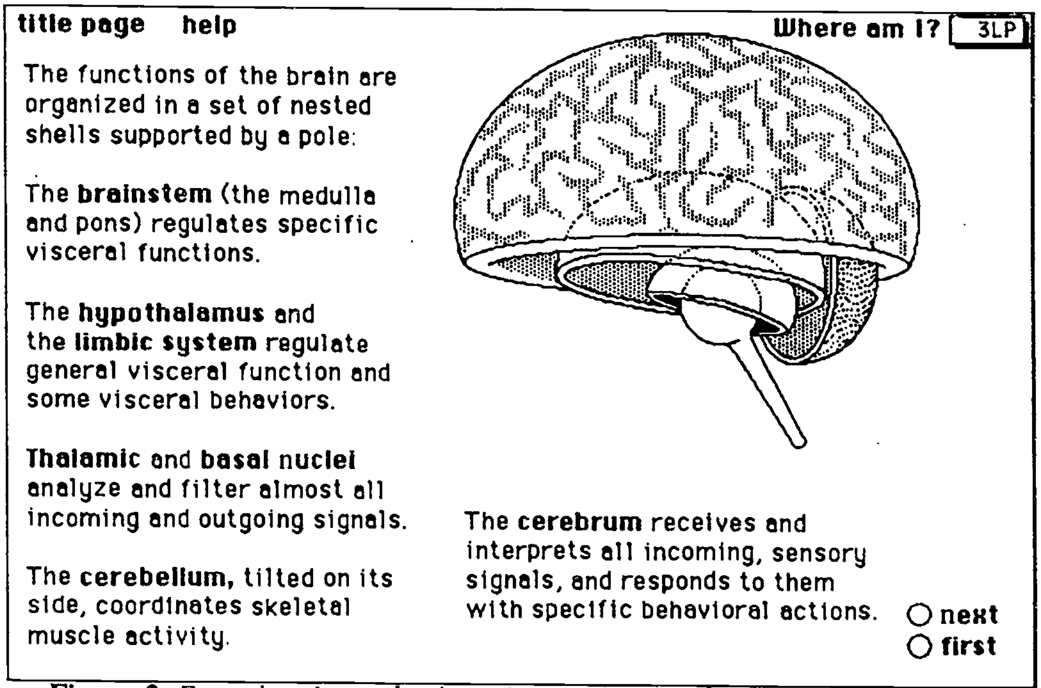


Figure 2. *Fostering the student's active involvement by identifying brain parts*

Movies of this nature enhance the first two of the three spatial abilities discussed above. Spatial perception, which is the ability to determine spatial relationship with respect to the orientation of one's own body, is obtained by the different viewing angles. Mental rotation, i.e., the ability to mentally rotate a two- or three-dimensional figure, is experienced by the student watching the brain movie. The student may, at any point along the movie, "freeze" it by simply moving the cursor away from the bottom area of the movie display area, and "unfreeze" it by moving the cursor to the same area. As long as the cursor is there and the mouse rests, the movie repeats itself. This utmost flexibility in movie control is accountable for obtaining high spatial understanding, as the student can stop the movie and advance it at any desired point to examine details that seem not to be clear enough.

Linking Structure with Function: the Barrier Test

The studyware offers the student a number of options to evaluate himself/herself. This is done through voluntary quizzes (Dori & Yochim, 1990) and mandatory tests. Before the student is allowed to advance from the introduction to the second topic, Sensory Input and Motor Output, the Brain module asks the student to try to passe a mandatory barrier test. The test is composed of a set of five questions chosen randomly by the program, ensuring that the same test is not administered to the same student more than once. This is done by keeping a record of the tests given to each student, who is required to identify himself/herself before the test is presented.

Figure 4 is a sample of a barrier test of the Organization and Brain Function sub-topic. It is aimed at testing the extent of meaningful learning the student obtained. The five problems presented in the test are covered by a field saying **Click here for question i**, with i = 1,...5. In Figure 4, the student has already answered questions 1, 2, and 4. He scored 61% in question 2 and lower than a passing grade (set at 55%) in question 4. The low achievement in question 4 is responded by the studyware's advise to review the specific sub-topic *Surface Brain Structures*, as the question deals with this sub-topic. Question 5 has just been exposed but not yet answered. The questions are oriented towards several

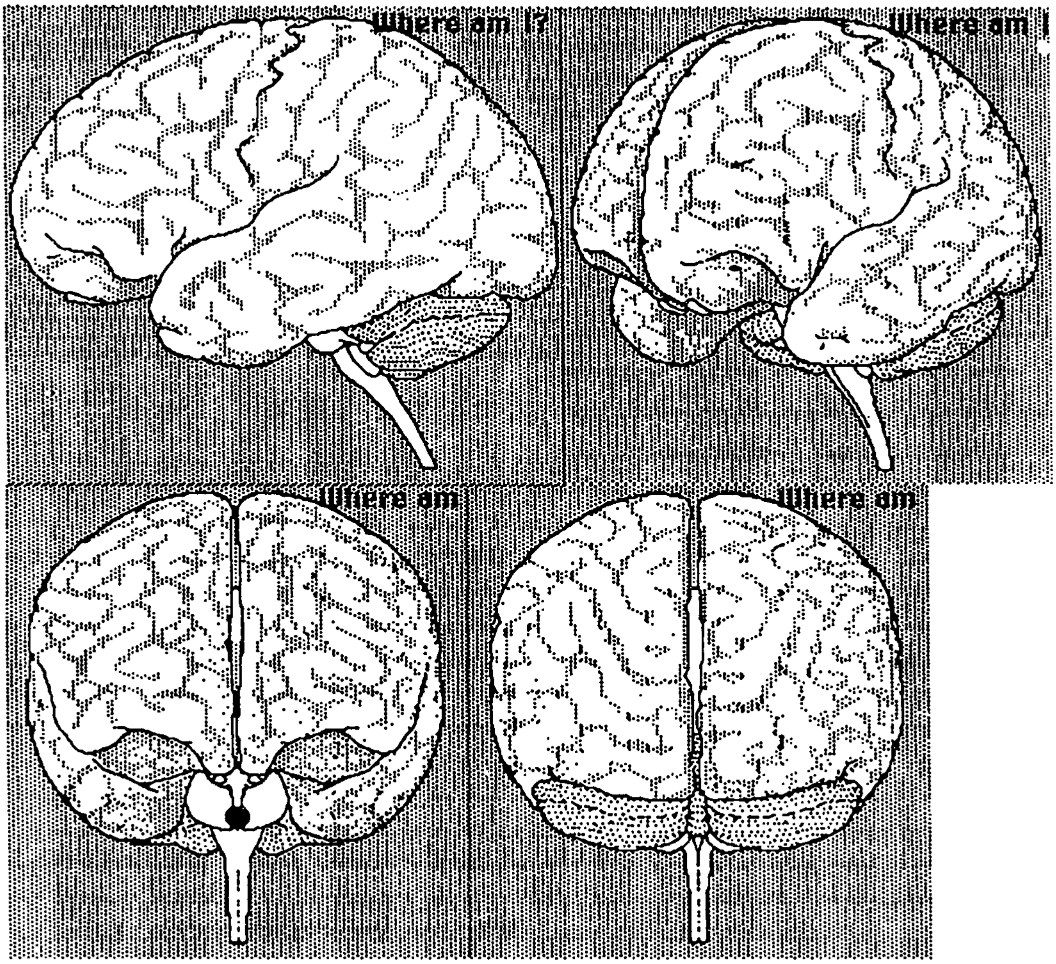


Figure 3. *Sample pictures of the brain structure movie*

cognitive levels. Question 1 requires the identification of a single part (basal nuclei). Question 2 asks to identify three parts in the brain structure: hypothalamus, mammillary body, and fornix. Question 4 concerns understanding of the spatial brain structure: it calls for pointing at a major subdivision of the cerebrum. Correctly answering this requires that the student understands that the cerebrum consists of several major parts and there are four points that need to be pointed at. This question reflects the level of the third spatial ability, spatial visualization. It involves problem solving of tasks with manipulations of spatially presented information. Questions 3 and 5 requires comprehension of the relation between structure and function. Higher levels of problem solving abilities appear in the barrier tests before entering the third topic - Processing the Signals. When the student is done with answering the five questions, clicking on the **grade** button sends a message to the program to calculate the grade. A new screen is then opened, telling the student his/her grade. Having passed the test, the student is able to advance to the next topic. Otherwise, a list of sub-topics that need be reviewed shows up below the grade with buttons leading directly to each sub-topic. As can be seen, the system monitors the students' performance as expressed by their responses to problems posed to them and reacts accordingly. This is a first step in imitating the role of an exemplary teacher, who is expected not only to grade the test but, more importantly, provide the right directions for the appropriate remedial learning activity.

Summary

We have presented a multimedia-based studyware application, which serves as an evolving model for an ICAI shell. With the resulting study modules, students can engage in enjoyable active learning and enhance their spatial abilities while the system monitors their responses and reacts accordingly.

The enabling technology of multimedia suits the non-linear nature of knowledge, which is so difficult to convey by books. The Brain module of the *Human Physiology* studyware has been presented as a case in point. It is our belief that the concepts and architecture proposed in this work will be further developed and take advantage of the synergy emanating from the fusion of modern educational theories and the opportunities offered by multimedia technology.

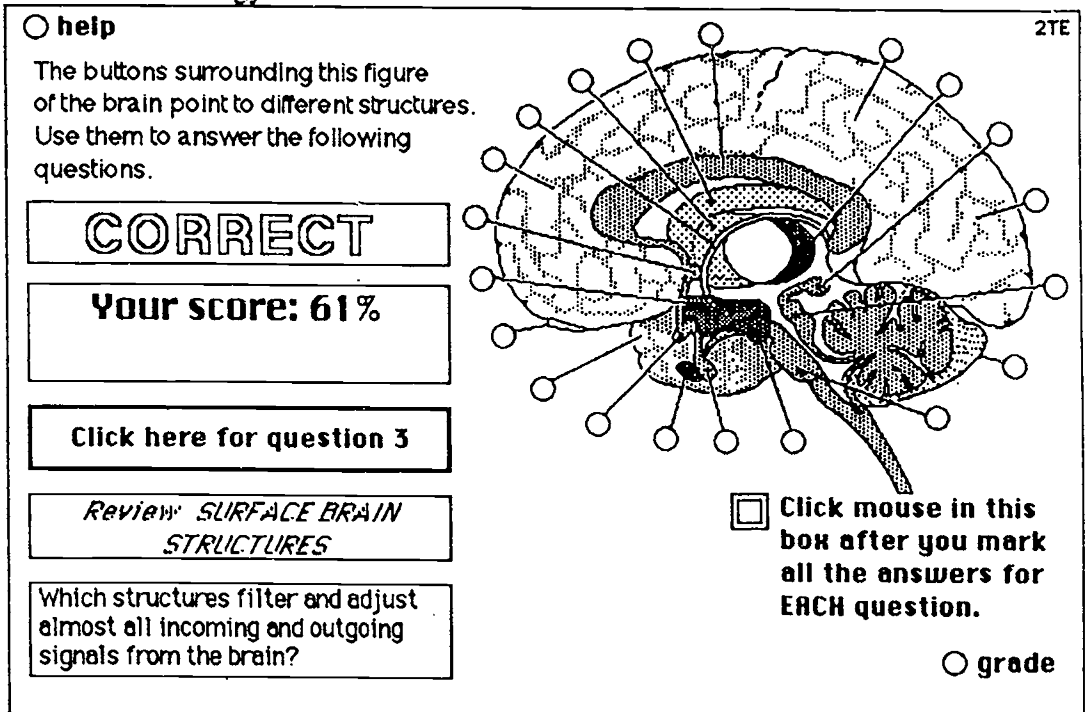


Figure 5. *A barrier test sample of the Organization and Brain Function sub-topic*

References

Ben Haim, D., Lappan, G., & Houang, R.T. (1988). The Effect of Instruction on Spatial Visulaization Skills of Middle School Boys and Girls. *American Educational Research Journal,* **25**, 1, 51-71.

Dori, Y.J., Dori, D. &Yochim, J.M. (1992). Characteristics of an Intelligent Computer Assisted Instruction Shell with an Example in Human Physiology. *Journal of Computers in Mathematics and Science Teaching*, **11** (3/4), 289-302.

Dori, Y.J. &Yochim, J.M. (1990). Learning Patterns of College Students Using an Intelligent Computer Assisted Instruction Module. *Journal of College Science Teaching*, **20**, 2, 99-103.

Linn, M.C., & Peterson, A.C. (1985). Emergene and Characterization of Sex Differences n Spatial Ability: a Meta-Analysis, *Child Development*, **56**, 1479-1498.

Lord, T.R. (1985). Enhancing the Visuo-Spatial Aptitude of Students. *Journal of Research in Science Teaching*, **22**, 5, 395-405.

Lord, T.R. (1987). A Look at Spatial Abilities in Undergraduate Women Science Majors. *Journal of Research in Science Teaching*, **24**, 8, 757-767.

McGee, M.G. (1979). Human Spatial Abilities: Psychomotoric Studies and Environmental, Genetic, Hormonal and Neurological Influences. *Psychological Bulletin*, **86**, 889-918.

Small, M.Y., & Morton, M.E. (1983). Spatial Visualization Training Improves Performance in Organic Chemistry, *Journal of College Science Teaching*, **13**, 1, 41-43.

Yochim, J.M., & Dori, Y.J. (1993). *Human Physiology, Interactive Tutorial and Workbook,* West Publishing Company, St. Paul, MN.
A– *Nervous System: The Brain;* B– *The Endocrine System;* C– *The Digestive System.*

Team Training Shell: a Groupware, Multimedia-Supported Application Generator

D. DORI, M. ALON, AND Y.J. DORI

Faculty of Industrial Engineering and Management
Technion, Israel Institute of Technology, Haifa 32000, Israel

Abstract

Since real-life situations of trauma training are practically not available, a proper substitute must take advantage of the most recent advents in multimedia and groupware technologies. Multimedia visualization is of particular importance in trauma training, as the crucial step of the patient's initial assessment is largely based on surface check. Using trauma team training as a case in point, our long-term goal is to design a domain-independent Team Training Shell (TTS)—a generic scheme for team training application generator. It is expected to enable the creation of training executables in any domain that involves the need for coordination among team members charged with a common mission. The design of the TTS architecture requires the integration of concepts and techniques from multimedia-supported human-machine interaction environment, groupware-based team collaboration, networking and distributed applications and databases. Our work is oriented towards obtaining synergy through integrating the benefits of groupware with multimedia. In this paper, we describe the background and considerations involved in the TTS design by using the trauma team training as a case in point.

Multimedia as a Supporting Technique

Most existing software systems support the interaction between the user and the system, but usually do not support interaction among users. We focus on the class of coordination systems, that allow individuals to view their actions, as well as the relevant actions of others, within the context of the overall goal. The main conceptual components of a coordinated groupware system are the following [3]: **Shared context**: A set of objects and actions performed on these objects that are visible to a set of users. **View**: A multimedia representation of some portion of a shared context. **Role**: A set of privileges and responsibilities attributed to an agent. **Coordination management**: A mechanism for coordinating simultaneous operations.

It is within the View component of that multimedia is heavily involved. The contribution of multimedia is especially important in the case of training systems, as these systems require human cooperative responses to simulated real-life situations displayed to them. Multimedia combines elements of motion and still video images, special effects, synthetic video, fast graphics and text with the interactive capabilities of a desktop workstation. Although most of the research in this area centers on individual multimedia applications that use stand-alone workstations, great potential for multimedia lies in the wide-spread development and use of distributed multimedia applications [1]. There is a number of existing experimental multimedia-based systems. Among them are EDUCOM [5], and IRIS Intermedia system [4]. Our work is oriented towards obtaining synergy through integrating the benefits of groupware with multimedia, while alleviating the burden of taking care of the complex management, control and coordination problems discussed above. The tool that we propose for this purpose

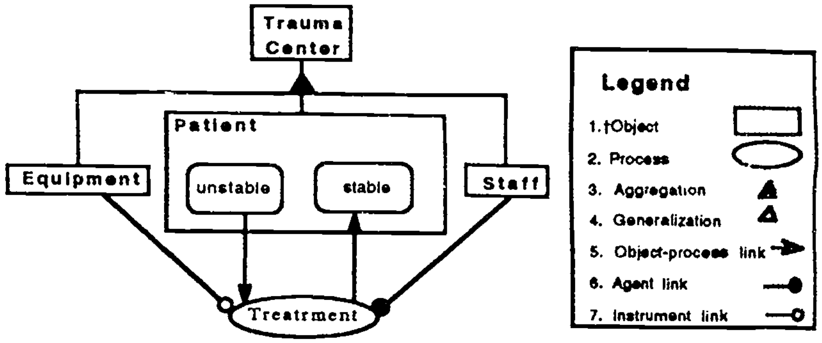


Figure 1: An OPD of a Trauma Center

is the Team Training Shell (TTS). Before going into the details of the shell, we introduce the Object-Process Analysis (OPA) methodology that we use as a tool for designing the TTS.

Object process Analysis

The basic observation underlying the Object Process Analysis (OPA) [2] approach is that every thing in the universe of interest can be classified either as an object or as a process. The approach combines Object Oriented Analysis (OOA) to represent the static-structural aspects, and the Data Flow Diagram (DFD) to represent the dynamic-procedural aspects, into one integrated representation approach, Object-Process Diagram (OPD). Figure 1 is an OPD of a trauma center. A trauma center consists of three main objects: Staff, Equipment and Patient, depicted as rectangular boxes, and a Treatment process, shown as an ellipse. The object treated by the process is an unstable patient and the desired outcome is a stabilized patient.

As shown in the legend of Figure 1, an Object-Process link leads from objects to processes and vice versa, describing the input and output objects needed to maintain the process. Blank and solid disks terminating a line, denote instrument and agent links, respectively. An instrument link connects the object which is needed to carry out the process at which it points. Likewise, the agent link connects the object (usually person), which carries out the process at which it points. Solid and blank triangles denote the two basic structural relations whole-part and generalization-specialization, respectively.

The Team Training Shell (TTS)

The Team Training Shell (TTS) is a multimedia-supported framework for developing training modules for teams who work in coordination to achieve a common goal. The system in figure 2 consists of three main processes: Training Authoring, Training Execution and Evaluation. The main purpose of TTS is to transfer the Team from a base-level proficiency to an improved one.

The Team Training Authoring process is carried out by the Team Training Author, who has the Domain Knowledge and uses TTS with its Supporting Environment as the authoring instrument. In this authoring process, the domain expert generates an Executable Training Module which has a Training Scenario Tree. A node in this tree represents a decision point while each edge represents the relative weight assigned to the selected action.

The resulting Executable Training Module is used as input to the Team Training Execution process. This process is supervised by the Team Trainer, who is a domain expert that has Domain Knowledge. The Supporting Environment is an integrated server combining a Multimedia interface, a Communication unit, an extended Database management system and a

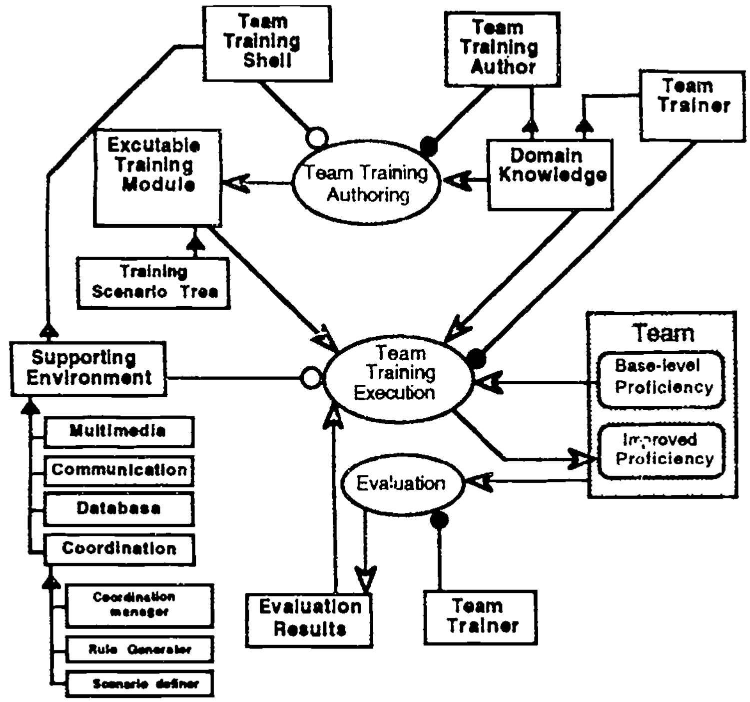


Figure 2: Team Training Shell - conceptual overview

Coordination management unit. It provides both the Authoring and the Execution processes with the appropriate tools. Among them is the multimedia tool kit, which allows selection of the suitable media combination, the activation time and duration. The Evaluation process provides ongoing follow-up facilities. Its important constituents are performance history logging, comparison with previous/standard results, and evaluation techniques.

Multimedia in TTS

A Team Training Shell, like many other shells, should hide the implementation details, providing the author with a convenient tool for creating the target application. The TTS architecture is based on a number of distributed workstations, one for each trainee, connected through a high-speed communication network. Each trainee sees and operates the system from his/her viewpoint. Concurrently, the trainer can show a global view of the whole system to each trainee or to all of them. The trainee is supposed to follow the scenario activated by the trainer and respond as s/he is expected.

A typical screen display comprises several independent views: The trainee view, the integrated general view, general common information view and a dialog-box for menu-driven interaction, through which the trainee performs the simulated actions and communicates with his peers. The trainee's view is extracted from the integrated view and can be manipulated by each trainee for himself. Normally, this view is used for showing the trainee a video clip that represents the simulated reality at that moment. A typical training team includes team members, one of whom is designated as the team head. A regular team member may look at his own view, which displays his role and past performance. The Team Head is the member who is in charge of the team and instructs the other team members. In the case of trauma treatment teams this is the chief physician. In a tank crew this is the tank commander, etc. The Team Head, like the trainer, has the option of looking at an integrated view of the session and the capability of monitoring and following each of the member's view at any time. He may also issue commands for session flow changing or problem solving.

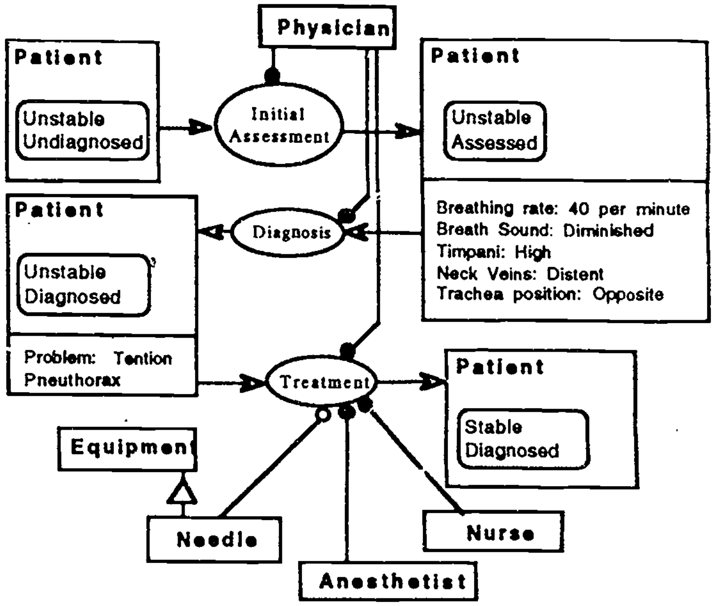


Figure 3: The Treatment process of a trauma patient with Tension Pneuthorax

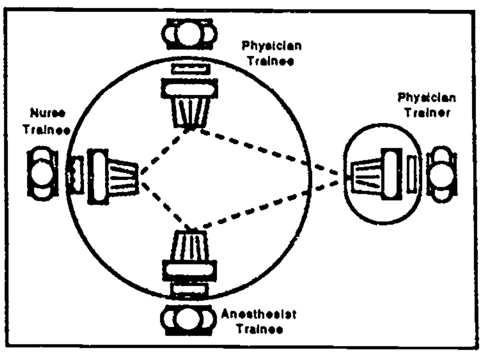


Figure 4: An overview of a team training setup

A Trauma scenario example

The treatment process begins with an initial assessment. This is followed by the definition of the immediate problems causing the stabilization interference. It ends with an application of the appropriate treatment. The initial assessment is divided into the following check sequence: Airway, Breathing, Circulation and Disability (known as "ABCD").

The problem addressed in this example is *tension pneuthorax* . The trainee should diagnose it from the description and data provided in the scenario.

Figure 3 is an "explosion" (procedural scaling-up) of the Treatment process described in figure 1, showing three sub-processes: Initial Assessment, Diagnosis and Treatment.

A trauma patient is evaluated by the three attributes: Assessment (unassessed / assessed), Diagnosis (undiagnosed / diagnosed), and Stability (unstable / stable). Each attribute combination is a different *status* of the patient. The mission of the team can now be formulated as changing the patient status from (unassessed, undiagnosed, unstable) into (assessed, diagnosed, stable).

Figure 4 shows a set-up of a team consisting of a Trainer Physician and three trainees:

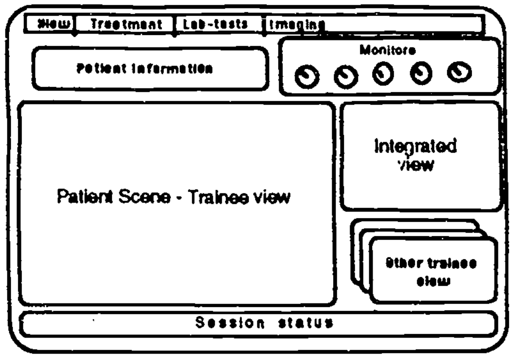


Figure 5: TTS trainee interaction screen

Physician, who is the Team Head, Nurse, and Anesthetist. Each of the four members is equipped with a workstation through which s/he sees the scene, communicates with the other team members, and performs the required activities.

Figure 5 depicts the layout of each trainee interaction screen. The main part of the screen is devoted to the trainee specific view. Another frame displays the team view which the trainer usually sees. A view usually displays a video movie representing the scenario as it proceeds. It constantly changes as the session proceeds, and can be controlled by the trainer. The trainee may ask to perform video operations such as switching the specific view with the team view and zooming part of the view. Another part displays the current monitor results, such as blood pressure, beep rate, breathing rate etc. The status bar is used for session orientation, it shows the current session stage, the time left and performance evaluation. In addition to the permanent display portions, the trainee may use a menu to see other trainees view, select and activate applicable operations, and display lab results and electronic images.

The training session begins with a video movie presenting a patient breathing spontaneously, but with a very high rate. The scenario is set so as to allow the physician to detect the patient's parameters by displaying the patient in various positions and angles and viewing the data displayed by the monitoring equipment. The trainer may slow or even stop the running scene and rewind the scenario if deemed necessary. Consultation among team members is enabled either by directly addressing each other or issuing a consultation request through the system.

In our example, the physician is expected to acquire the data concerning the patient's breathing rate. If he does so within the expected time frame, the system responds with the (exceptional high) value of 40 beats/minute. Otherwise, the patient's condition continues to deteriorate as the session proceeds. Using the knowledge of very high breathing rate, the physician is now expected to watch and, using audio capabilities, listen carefully to detect findings in breath sound, percussion, significant distention of neck veins and changes in the position of the trachea. Following this assessment process the physician is expected to obtain the following results: the patient has a diminished breath sound in the left side, left hyper-timpani, distention of neck veins, and opposite trachea position. The physician should be able to determine the problem. The number of possible attribute value combinations grows combinatorially. It makes the diagnosis process highly complicated. At any decision point the trainee has a number of possibilities from which he has to select and proceed. The score of each action determines whether the system allows the scenario to proceed or else prevents it while issuing an appropriate explanation. Normally there is one recommended option in the scenario tree. Some of the possible actions are completely wrong, as they lead to catastrophic results. The system or the trainer may prevent such actions from being selected by any trainee. Assuming the correct assessment has been selected, the physician determines the problem and notifies the team of his diagnosis along with the treatment to be performed. Then, the physician asks the nurse to prepare the patient for a needle insertion to the chest. The nurse is supposed to perform a series of preparatory activities: undressing the patient, cleaning the chest, supply equipment, etc. This is done by selecting operations from a list of possible operations displayed through the menu bar, and applying them to the patient's body by pointing to the appropriate body location. Each applied operation that has a visual effect on the presentation is displayed, either by moving to the appropriate video frame or by combining a still picture with the video movie. After the patient is ready for the needle insertion, the physician selects the needle size from a displayed list or, points to the exact location and then inserts the needle. As a result of this treatment, the patient's condition is stabilized within the appropriate time span.

Conclusion

We have introduced the concept of Team Training Shell (TTS) and its components. Multimedia technology and coordination management are two key components of TTS. There are many theoretical and technological details in the supporting environment that may not yet be ripe for the required application we have described. A major obstacle is the lack of tools that support concurrent distributed multimedia display and control. To prove that the implementation of a team training executable is at all a feasible task, we expect, as a first stage, to accomplish the analysis, design, and implementation of a prototype that demonstrates the concept.

Acknowledgement
This work was supported by the Technion Research Fund, No. 231-219.

References

[1] Bellcore Information networking research laboratory (1993). The Touring Machine System. *Communications of the ACM, 36(1), 68-77.*

[2] Dori,D.,Phillips,I.,and Haralick,R.M.(1993). Incorporating documentation and inspection into computer integrated manufacturing: An object-process approach. *Applications of Object Oriented technology in manufacturing, Chapman-Hall, London.*

[3] Ellis,C.and Rein,G.(1991). Groupware: The Research and development Issues. *Communications of the ACM.*

[4] Haan,B.J.,Kahn,P.,Riley,A.,Coombs,J.H.,Meyrowitz,N.K.(1992). IRIS Hypermedia Services. *Communications of the ACM, 35(1),36-51.*

[5] Hoffert,M.,Gretsch,G.(1991) The Digital News System at EDUCOM: A convergence of interactive computing, newspapers, Television and High-speed Networks. *Communications of the ACM,34(4),113-116.*

Object-Process Analysis of Intelligent Computer Assisted Instruction Shell: the Polymer Courseware - a Case in Point

Yehudit J. Dori
Department of Education
in Technology and Science

Dov Dori
Faculty of Industrial
Engineering and Management

Technion, Israel Institute of Technology
Haifa 32000, Israel

Abstract: Recent advances in information technology have made feasible the development of intelligent computer assisted instruction study units. Analysis and design of such systems require the active involvement of a development team consisting of domain experts, educators and knowledge engineers. Science and technology teaching methodologies are frequently domain-independent. The important component that changes is the domain knowledge. This is the basis for the analysis and development of an Intelligent Computer Assisted Instruction (ICAI) shell. We apply an object-process analysis analysis approach to analyze the shell, the users, the system's modules and the relationships among them. The object-process diagram is used as a tool to visualize the relationships between objects and processes in the ICAI Shell. To exemplify some of the concepts, we provide a case study of a courseware development process of a CAI Polymer unit.

Theoretical foundations of the teaching/learning process on one hand and recent advances in multimedia technology on the other hand, have made possible the creation of a learning environment that benefits from both. Science and technology teaching methodologies are frequently domain-independent. The important component that changes is the domain knowledge. This is the basis for the analysis of a shell for Intelligent Computer Assisted Instruction (ICAI) development. The division of the learning environment and science or technology domain of study, into two distinct components of a studyware opens the way for the development of a generic framework that supports studyware authoring (Dori, Dori and Yochim, 1992). Such a framework, or "shell," should enable the expert team, charged with the curriculum development, to concentrate on a sound presentation and teaching of the subject matter. At the same time, the shell is supposed to take care of putting the subject matter in the right educational context, relieving the developers from the burden of low level programming such as graphic user interface and repetitive tasks. The approach of a shell architecture is coherent with the reusability principle of software engineering: rather than "reinventing the wheel," software development should capitalize on existing code modules that have been extensively tested and proven to be reliable and efficient. Applied to our objective, on top of all this, education principles embedded in the shell would ensure a good starting point for ICAI developers.

An ICAI shell is a very large development project. A project of this order of magnitude requires a well founded methodology of systems analysis, design and implementation. This work uses an object-process analysis (OPA) approach (Dori, Phillips and Haralick, 1994) to visualize and analyze the shell, the users, the system's modules and the relationships among them. The OPA approach combines the object-oriented analysis (OOA) approach (Jacobson et al., 1992; Shlaer and Mellor, 1992), to represent the static-structural aspects, and the Data Flow Diagram (DFD) proposed by De Marco, (1978) and Yourdon and Constantine, (1978) to represent the dynamic-procedural aspects, into one integrated representation approach. The basic observation underlying the object process approach is that every thing can be classified as an object or as a process.

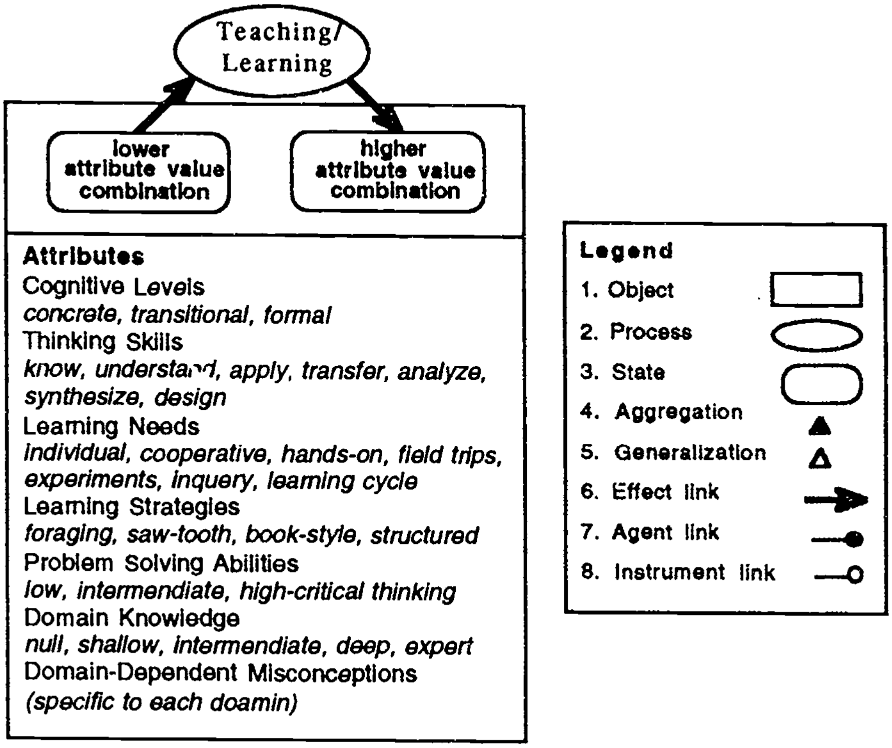


Figure 1. *An OPD showing the student's attributes and their possible values*

An Object-Process Diagram (OPD) is a diagram that depicts the classes within the system and the structural and procedural relations among them. The OPD uses a number of symbols, shown in the legend of Figure 1. One is the effect link, which leads from objects to processes and vice versa, describing the input and output objects needed to maintain the process. Another pair of symbols are blank and solid disks terminating a line, which denote instrument and agent links, respectively. An instrument link connects the object which is needed to carry ou' the process at which it points. The agent link connects the object (usually person), which carries out the process at which it points. Finally, the solid and blank triangles denote the two basic structural relations whole-part and generalization-specialization, respectively. The whole-part relation denotes aggregation of one or more classes to a higher-level class. The generalization-specialization (gen-spec, or "is-a" relation) denotes the specialization of a higher-level class.

The Student and the Teaching/Learning Process

The end user – the student – is modeled as an object affected by the teaching/learning process. Each student can be characterized by a set of attributes which, at a given point in time have a particular combination of attribute values. The major student's attributes which are of interest for an educator are listed in Figure 1. The possible values for each such attribute are listed beside each attribute. Where applicable, they are arranged in an increasing order of advancement and/or sophistication. A student may, for example, have a concrete Cognitive Level, with Thinking Skill *understand*, with *individual* and *hands-on* Learning Needs, with book-style Learning Strategy, low Problem Solving ability, shallow Domain Knowledge, and Domain-Specific Misconceptions X and Z. This attribute value combination is one state in the state space. The state space is the collection of all the possible attribute value combinations a student may assume. The purpose of the Teaching/Learning process is to transfer a student from a state with a given attribute value combination to another state, in which the overall attribute value combination is higher. The most common improvement is expected to occur in the value of the Domain-Specific Knowledge, but, at least in the long run, Teaching/Learning process is expected to yield improvements in attributes such as Thinking Skills, Problem Solving Abilities and Learning

Strategies. Figure 1 is an Object-Process Diagram (OPD), in which Student is the object, represented as a rectangle with a state-transition diagram at its upper compartment, consisting of two states, represented as rounded-corner rectangles, and with the list of attributes and their possible values at the lower compartment. The Teaching/Learning process, depicted as a bubble (ellipse), is shown changing the student's state from lower to higher attribute value combination. With the above definition of the student's states the Teaching/Learning process, we proceed to describe an overall concept of the ICAI Shell. it consists of the Student Module, the Expertise Module, and the Teaching/Learning Module.

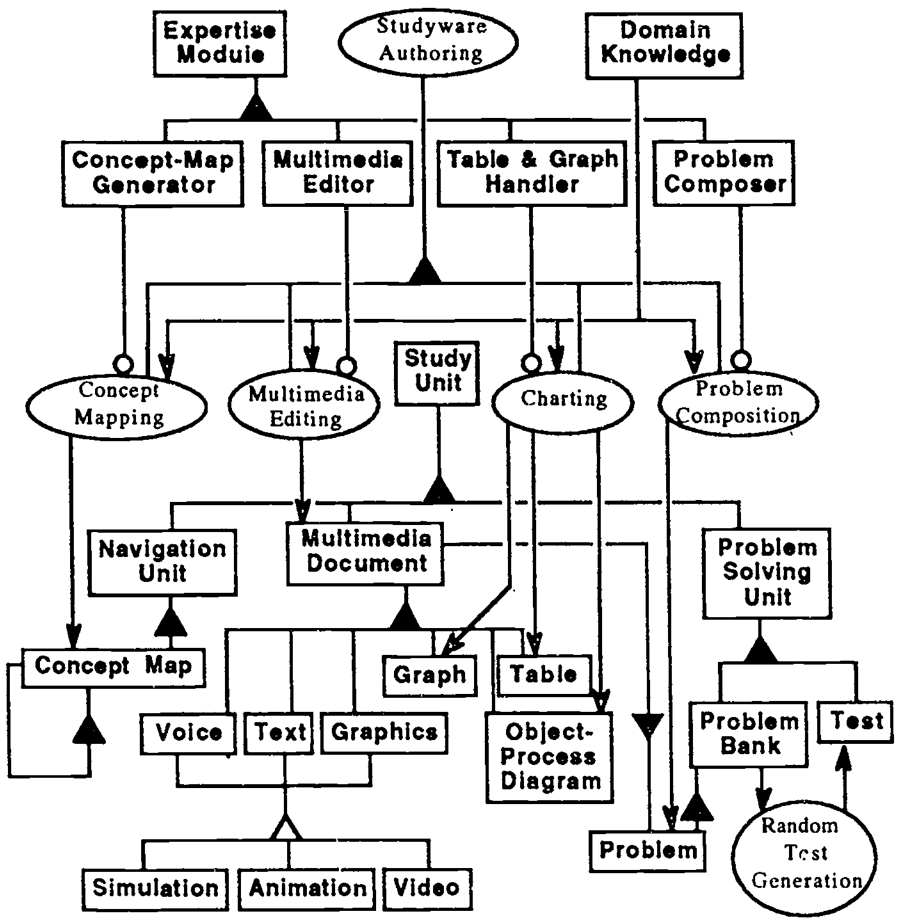


Figure 2. *An OPD of the studyware authoring process and the resulting study unit*

The Expertise Module Object within the ICAI Shell

The Expertise Module object class and the Studyware Authoring process class are shown in Figure 2. The Expertise Module consists of four objects: the Concept-Map Generator, the Multimedia Editor, the Table&Graph Handler, and the Problem Composer. The Concept-Map Generator is a tool for the process of Concept Mapping – a powerful means for the representation of knowledge at various levels of depth. It enables easy and reliable construction of concept maps that are linked to each other as described by Fisher (1990). Computerized concepts maps may enable navigation around the study unit and, possibly, to and from other study units. It also helps students in putting concepts in the right context, thereby organizing pieces of knowledge in a meaningful way. An advanced option of this generator can be its ability to draw Object Process Diagrams, such as the figures in this work. Thus, concept maps in the ICAI Shell environment use the notation and semantics of the object-process paradigm, yielding an enhanced version of concept maps. The Table&Graph Handler is an electronic spreadsheets that enables making tables and graphs and define constraints among table cells. It is used for Charting and produces Graphs and Tables. One example, described by Linn and Songer (1991), makes use of a similar tool for teaching thermodynamics for middle school students to help them construct more abstract and general scientific principles at a macroscopic level.

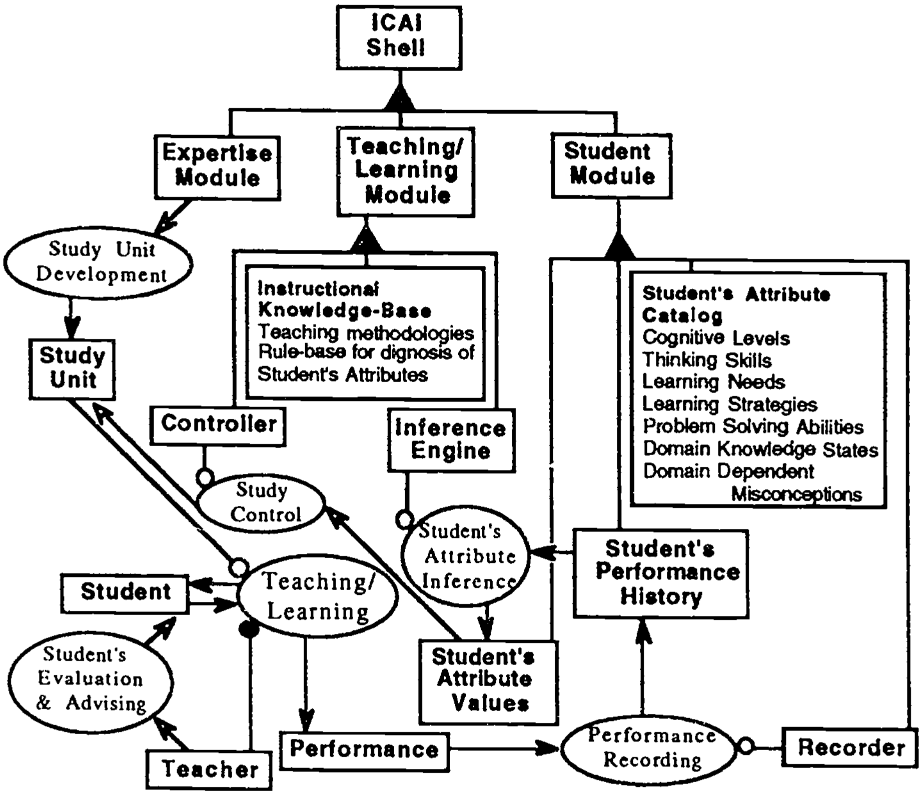


Figure 3. *The interaction between Student, Teacher, Study Unit and the ICAI Shell*

The Problem Composer is the fourth component of the Expertise Module. It has a variety of problem templates for various kinds of problems, such as push-button multiple choice combinations. It is used in the process of Problem Composition which results in Problems for the Problem Bank (Dori and Yochim, 1990). Each problem is characterized by the topics it is designed to test, the template used and the degree of difficulty. The Studyware Authoring process then consists of Concept Mapping, Multimedia Editing, Charting, and Problem Composition. The result of these processes is the Study Unit, which, in turn, consists of a Navigation Unit, a Multimedia Document, and a Problem Solving Unit. The Navigation Unit is responsible for orientation of the student in the web of screens so that he or she does not get lost and can always find out "where am I".

The Multimedia Editor is a composite software component for multimedia document authoring. The Problem Solving Unit consists of a Problem Bank and Tests which are generated by randomly selecting problems with the appropriate attributes. Tests and quizzes are characterized as being either voluntary or mandatory. This mechanism ensures that the student does not browse aimlessly throughout the entire module and across modules before obtaining a minimal level of knowledge about the subject matter he has studied (Yochim and Dori, 1993). The Expertise module is one of the three modules of the ICAI Shell, the other two being the Teaching/Learning Module and the Student Module (see Figure 3). The Student Module contains a Student Attribute Catalog with the attributes and values, described in Figure 1. For each individual student, this module keeps record of Performance History and Attribute Values. The Recorder monitors and records students' responses of interest, such as the order and duration of viewing cards, answers to quizzes and tests, etc. The Teaching/Learning Module is the heart of the ICAI Shell. It has an Instructional Knowledge Base with Teaching Methodologies and a Rule-Base for diagnosis of students' attributes. It has a Controller and an Inference Engine that, based on the Student's Performance History and current Performance, infers and updates the current Student's Attribute values. These are used for the process of Study Control which is done by the Controller. It controls the learning pace, depth and abstraction level. The Student is engaged in the process of Teaching/Learning in which the Study Unit is the instrument and the human Teacher is the agent. Using data from the Student Module regarding present and past performance, the teacher also evaluates the student and advises for effective learning strategies.

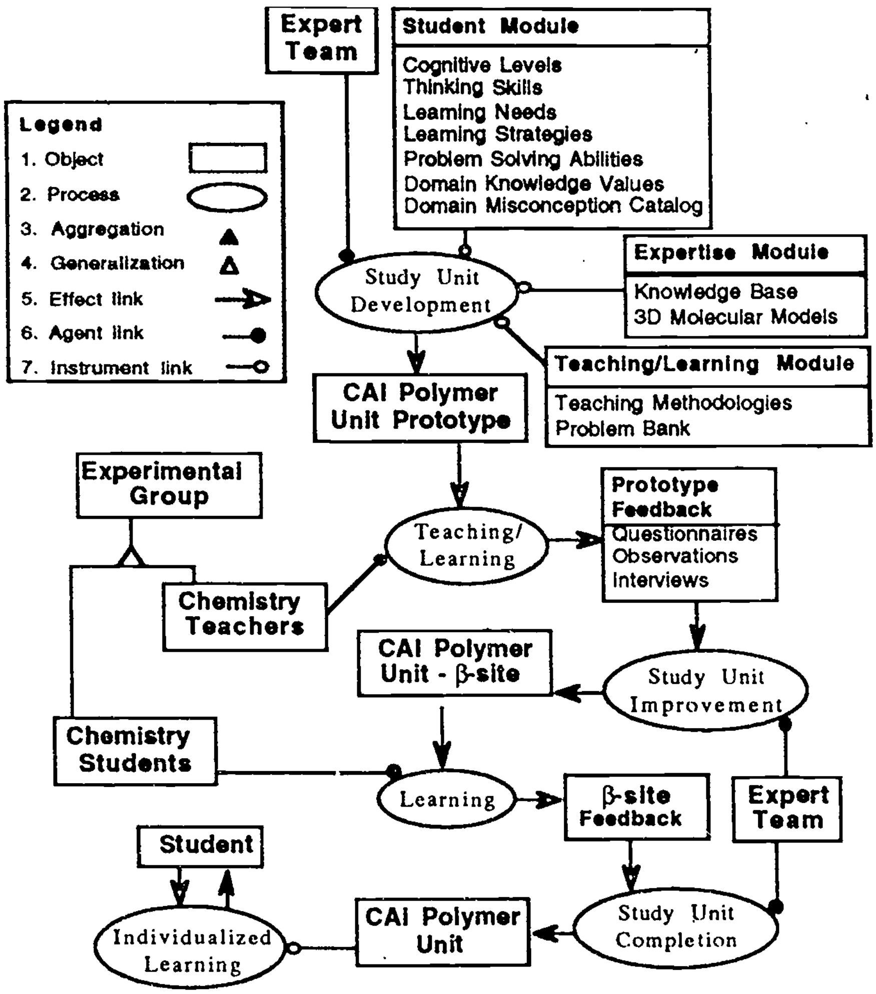


Figure 4. *An Object-Process Diagram of the Polymer Courseware Authoring*

The Polymer Courseware Authoring: a Case in Point

To exemplify some of the concepts introduced in the previous sections, we provide an example of a complete courseware development process of a Computer Aided Instruction Polymer unit (Dori and Barnea, 1993). Figure 4 is an OPD of the CAI Polymer Unit development. The agent that authors the courseware is the expert team, which includes experts in the area of the chemistry of polymers and science education. The products (objects) of this process are the Student, Expertise, and Teaching/Learning Modules. As a preliminary activity, the expert team analyzed the Students' Attributes and Teaching Methodologies relevant to the particular subject matter - the study of polymers. For example, a major learning need has been found to be understanding the properties of macromolecules that are related to their three-dimensional structure. Accordingly, one teaching methodology was set to be 3D Molecular Modeling. The outcome of the Study Unit Development Process was the CAI Polymer Unit Prototype. The Prototype was used by an Experiment Group of Chemistry Teachers who took part in an in-service training aimed at both introducing the teachers to the Unit and obtaining feedback on the prototype. The Prototype Feedback, which was compiled from a series of (pre- and post-training) Questionnaires, Observations and Interviews, was used by the Expert Team for the process of Study Unit Improvement, which resulted the β-site CAI Polymer Unit version.

Students who major in chemistry were the Experiment Group for the β-site version, and the β-site Feedback obtained form them was the input to the Study Unit Completion. The end product – the CAI Polymer Unit – is the object used as an instrument for the Individualized Learning process by the Student. This process potentially changes the student's attribute values from a lower to a higher combination.

Summary

We have presented an Object-Process analysis of a system for developing courseware. With the resulting study units, students can engage in active learning while the system monitors their responses and reacts accordingly, imitating the role of an exemplary teacher. This Object-Process analysis serves as an architecture for an Intelligent Computer Assisted Instruction Shell. It is based on a holistic view of the student as an object and the teaching/learning as a process that transforms him/her from a state of lower attribute value combination to a higher one. A key factor in this system is the enabling technology of multimedia, which opens the way for the creation of a multi-faceted learning environment and suits the non-linear nature of knowledge, which is so difficult to convey by books. A preliminary stage in the development of these concepts has been presented as a case in point by the CAI Polymer Unit Development Process. We believe that the concepts and architecture proposed here will be eventually materialized and take advantage of the synergy between the opportunities offered by modern educational theories and technological advances.

References

Dori, D., Phillips, I. and Haralick, R.M. (1994). Incorporating Documentation and Inspection into Computer Integrated Manufacturing: an Object-Process Approach, to appear in *Applications of Object-Oriented Technology in Manufacturing*, Ed. S. Adiga, Chapman & Hall, London.

Dori Y.J. and Barnea, N. (1993). A Computer Aided Instruction Module on Polymers, *Journal of Chemical Information and Computer Sciences*, **33**, 3, 325-331.

Dori, Y.J., Dori, D. and Yochim, J.M. (1992). Characteristics of an Intelligent Computer Assisted Instruction Shell with an Example in Human Physiology, *Journal of Computers in Mathematics and Science Teaching*, **11**, 3/4, 289-302.

Dori, Y.J. and Yochim, J.M. (1990). Learning Patterns of College Students Using an Intelligent Computer Assisted Instruction Module, *Journal of College Science Teaching*, **20**, 2, 99-103.

Fisher, K.M. (1990). Semantic Networking: The New Kids on the Block, *Journal of Research in Science Teaching*, **27**, 10, 1001-1018.

Jacobson, I., Christerson, M., Jonsson, P. and Overgaard, G. (1992). *Object-Oriented Software Engineering*, ACM Press, Addison Wesley, New York.

Linn, M.C. and Songer, N.B. (1991). Teaching Thermodynamics to Middle School Students: What are Appropriate Cognitive Demands? *Journal of Research in Science Teaching*, **28**, 10, 885-918.

Yochim, J.M., and Dori, Y.J. (1993). *Human Physiology – The Endocrine System*, Interactive Tutorial and Workbook, West Publishing Company, St. Paul, MN.

Yourdon, E. and Constantine, L.L. (1978). *Structured Design*, 2nd edition, Yourdon Press, New York.

HOED: Hypermedia Online Educational Database

E. DUVAL, H. OLIVIÉ
Department of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200 A, B-3001 Leuven (Heverlee), Belgium
Email: {Erik.Duval,olivie}@cs.kuleuven.ac.be

Abstract: This paper presents HOED, a distributed hypermedia client-server system for educational resources. The aim of HOED is to provide a library facility for hyperdocuments [Engelbart, 1990]. Its main application domain is education. The HOED database not only holds the educational resources themselves, but also data describing characteristics of these resources. Distinctive features of the HOED server are the separation of data regarding content (nodes) and structure (links), and use of a DataBase Management System (DBMS) rather than files for data management. We will explain how we make use of the HOED infrastructure for the development of private study modules on basic computer science topics and for re-use of courseware components. We will also briefly compare our work with related research.

1 Introduction

The work presented here deals with a *distributed hypermedia* system [CACM 1994]. It is related to and influenced by numerous fields of computer science, including information retrieval, databases, Computer Supported Collaborative Work (CSCW) and Computer Aided Learning (CAL).
In [Duval & Olivié, 1993], we presented a distributed *multimedia database* and explained how it can be used to manage multimedia and hypermedia educational resources. We will describe here how our work has evolved since then into a *client-server distributed hypermedia database*, called HOED (sections 2). The HOED server and its distinctive characteristics are discussed in section 3; the client environment is described in section 4. Two application areas are covered in section 5. After a comparison with related work in section 6, we conclude with a description of the current status and our plans for the future.

2 Architecture

The design of HOED is based on a client-server approach.

- The *server* takes care of data management and is itself distributed. A *hypermedia database* is responsible for management of the structural aspects. This hypermedia database is implemented on top of a *multimedia database* that takes care of content management, based on a mechanism that includes a type indicator (e.g. string, date, GIF image, etc.) with each multimedia value. Analogue and digital *raw data stores* offer physical storage facilities for the multimedia data.
- The *client* interacts with the end user or application program, initiates requests to the server, and displays the query results. In this way, the client hides details regarding the distributed nature of the server, the communication protocol and the database query language.

The design and implementation of the multimedia database and raw data stores has been covered in [Duval & Olivié, 1993]. The next section will present the server in more detail. Section 4 covers the client environment.

3 HOED Server

3.1 Data Model

The structural unit of data in HOED is the *node*. The primary function of a node is to contain multimedia content: text, computer software, audio or video streams, etc. In contrast, the primary function of a *link* is to contain structure (see also section 3.2).

An unlimited number of arbitrary (attribute,value) pairs can describe characteristics of nodes, links, or the participation of nodes in links. At any time, new (attribute,value) pairs can be added to the database. Some attributes are predefined, e.g. 'Type' that types nodes, links and participations of nodes in links. For particular types of these constructs, additional attributes can be predefined:

- We have e.g. defined a specific type of nodes for images on an analogue video disk about dentistry, produced at Bristol University (see also section 5). A predefined set of attributes describes the common properties of the material presented on these images: breed, sex, species, stain, etc.
- Specific types of links model specific types of spatio-temporal aggregation. 'Sequential aggregation' e.g. models a composite node as an ordered sequence of components.

3.2 Separation of Structure and Content

In conventional hypermedia systems (exemplified by e.g. Hypercard and Toolbook), content and structure are inextricably intertwined: links are embedded in document nodes. In HOED, links and nodes are separated for several reasons:

- *Maintenance* [Kappe, Maurer & Scherbakov, 1993] is accomodated by separation of structure and content. If e.g. a node is removed from the hypermedia database, then all links to the node that is being removed can also be removed, in order to avoid dangling references. This does not require a modification of any of the remaining nodes.
- Nodes and links can be *created, modified and deleted independently*: no conversion process is required to import new nodes, as the link information does not need to be imported into the document. This facilitates integration of separately developed tools: node editors e.g. don't need to provide link editing facilities. A new node can readily be integrated in the overall hypermedia network: new links can be added to existing nodes, so that these refer to the new node, without requiring a modification to the actual content of these nodes. An important consequence is that read-only documents (e.g. stored on a CD-ROM) can be linked to and from.
- When links are embedded in nodes, it is very difficult to support a *back-link capability* that identifies the links to a node [Engelbart, 1990]. A search through all nodes in the universe in order to identify those that contain a link to the node in question will be very time consuming. Incorporation of the links to a node in that node itself requires an update of the node whenever a new link to it is created or an old link is deleted. When links are stored separately, then the hypermedia database can simply be searched for all links pointing to a particular node.

3.3 Data Management

Some systems that do separate structure and content rely directly on a file system for data management purposes [Wiil, 1993]. Using a DataBase Management System (DBMS) for this purpose offers many advantages:

- The DBMS can take care of consistency, concurrency and access control, backup and recovery, data integrity and distribution.
- The query facility of the DBMS can be used to support:
  - *query based access*: This is especially relevant in a large scale networked hypermedia context, where navigation alone will no longer be appropriate if information overload is to be avoided [Duval & Olivié, 1992].

- *Intensional links* [Kappe, Maurer & Scherbakov, 1993]: Conventional links list explicitly the identity of the target node(s) of a link. Intensional links are defined by a set of search conditions, and refer to all nodes satisfying the conditions at the time of evaluation. They are therefore dynamic, in the sense that they can refer to different sets of nodes at different moments in time.
- *link filtering*: Search criteria can be defined to identify those links that are relevant to a particular group of users, e.g. in terms of user profiles that indicate their areas of interest, their level of expertise, etc. This is especially relevant in an educational context: e.g. once a student has obtained a sufficient grade in a particular test, more and more of a global hypermedia structure of that domain may be revealed to him.

4 HOED Client

We envision different kinds of clients accessing the HOED server. *Node editors* support creation, modification or deletion of multimedia node content. Typical node editors include CAD tools, authoring systems, text editors, etc. *Link editors* can be used to create links between nodes. A *browser* enables end users to explore the HOED information space, following links between related nodes. A *searcher* enables developers to specify search constraints that identify the resources they are looking for. The constraints can be composed of arbitrary complex combinations of simple constraints, using the boolean connectors AND, OR and NOT.
We are currently developing two sets of client components:

- A first set of components is under development in Tcl/Tk [Ousterhout, 1990], a user interface management system available from Berkeley. Tcl/Tk provides facilities for communication between applications, so that the different client components will be able to exchange messages. As indicated by the bottom arrow in figure 1, the client components interact with the HOED server using SQL, the standard query language for interaction with an RDBMS.

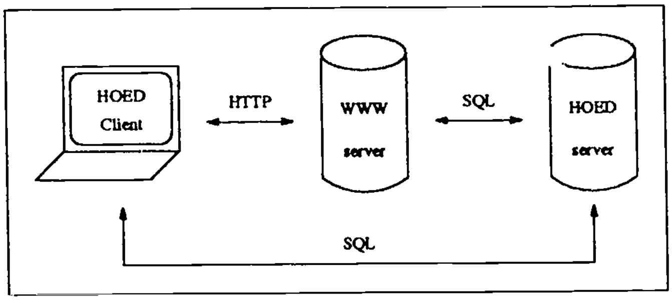


Figure 1: Gateway to the World-Wide Web

- Because of the enormous amount of documents available through the WWW, its rapidly increasing popularity and the free distribution of WWW software (both on client and server side, for many different platforms), we are developing a second set of client components, based on forms stored on a World-Wide Web (WWW) server (see also section 6) that interacts with the HOED server through a gateway.

  We rely on the Common Gateway Interface that defines a protocol for interaction between a WWW server and a non-WWW server. In this set-up, users access the HOED server through a WWW client (e.g. Mosaic or Cello) that interacts with the WWW server, using the HyperText Transfer Protocol (HTTP), as indicated on figure 1. The WWW server composes an SQL query for the HOED server, sends the query to that machine and presents to the end user the results that he receives. In this way, the WWW server is used as a "user interface server", as it holds the definition of the forms that define the screen lay-out and functionality during interaction. The more basic hypermedia storage and retrieval facilities are provided by the HOED server.

5 Application Areas

5.1 Basic Computer Science Modules

Since September 1993, we have started the development of a modular private study course on basic computer science topics for university students that do not need an in depth education in this field [Olivié & Duval, 1993]. The modular design of the course enables us to take into account the diverse background of the students and the peculiarities of the curricula in different faculties of the university. The modules will be stored as collections of linked multimedia nodes in HOED. Access to the modules

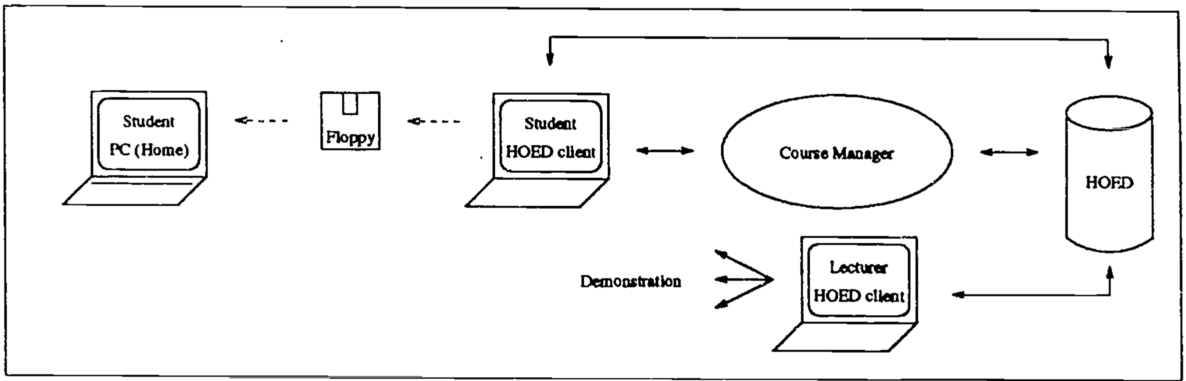


Figure 2: Basic Computer Science modules: Overview of Interactions

can proceed in different ways:

- A *course manager* is currently under development. This software component will support student evaluation and guidance. Pre-tests, based on intensional links (section 3.2) in terms of a student model, can refer a student to the most suited module.
- Bypassing the course manager, students can also navigate more freely through the learning resources in HOED, just as they can e.g. make use of the university *library*. This will enable them to explore the educational material, to pursue their own interests, or in order to find additional background material.
- A lecturer can retrieve (part of) a module from HOED in order to use it for demonstration purposes.

Both students and lecturers can access HOED over KULnet, the K.U.Leuven university computer network. Students are allowed to copy some of the educational material on floppy disk, so that they can use it on their own PC at home. This greatly increases availability of the resources for the students, the more so as a survey we carried out in 1993 among first year students in Engineering, Mathematics, Physics and Informatics showed that 82% of these students had access to a PC at home.

5.2 Re-Use of Courseware Components

In the CAPTIVE project ('89-'91) [Duval, 1992], aimed at 'Collaborative Authoring, Production and Transmission of Interactive Video for Education', we experimented with re-use of existing courseware components when developing new educational material. Video footage originally shot for a linear video e.g. was later on integrated in an interactive video. When trying to re-use existing resources, developers of new material face a number of problems. HOED provides at least a partial solution to three of these problems:

- Developers need to be able to find out *what* is available and whether they can use it for their own purposes. For the latter question, different considerations relate to the technical characteristics of the existing resources, copyright and pricing issues, the content presented in a (component of a) module, etc. This search can be supported by the query and navigation tools covered in section 4.

  Not only developers of new material can benefit from this facility: students or lecturers can also use it to access relevant material. In fact, providing this kind of access is the first aim of our collaboration with a team at *Bristol University*. In a first phase of this collaboration, ca. 250 images on a dentistry laser video disk have been described. The descriptions (including the location of the images on the disk) have been stored in the HOED server. When a relevant node has been identified, users can see the corresponding image, using a local tool for access to the video disk.

- Once the relevant resources have been identified, they can easily be accessed, using the client/server communication protocol. Developers need not be aware of *where* the material is physically located. In order to further refine the results of a query search, the nodes retrieved from the server can be used as a starting point for local navigation. in order to explore related material.

- Integration of existing resources is accomodated by the separation of node and link information: links can be created, modified and deleted *independently*. (see section 3.2).

The resulting mechanism for sharing of (parts of) nodes resembles Ted Nelson's idea of *transclusions*, or virtual copies [Nelson, 1993]. As described by Nelson, a mechanism can be established to compensate an author with a relatively small fee, every time one of his nodes is accessed. This approach may solve at least partially the copyright issue that often complicates re-use of independently developed resources.

6 Related Work

HOED can be compared with a number of recent client-server systems for networked information management over the internet [Obraczka, Danzig, & Li, 1993]:

- *Wide Area Information Servers* (WAIS) provide full text searching functionality with relevance feedback, i.e. the user can indicate relevant documents in an intermediate result, and WAIS locates similar documents that are likely to be relevant as well.
- *Gopher* provides access to information through a hierarchical menu system. Selection of a menu entry causes a sub-menu or document to be retrieved and displayed.
- The *World-Wide Web* (WWW) is an attempt at creating a world wide hypermedia system of linked documents. Links cannot only point to documents, but also to internet services such as remote login or file transfer.
- *Hyper-G* [Kappe, Maurer & Scherbakov, 1993] is a research project at the University of Graz.

WAIS and Gopher belong to computing paradigms (full-text search and hierarchical menu access respectively) that are fundamentally different from the hypermedia approach. The main disadvantage of the WWW is that links are embedded in documents (see section 3.2. As mentioned in section 4, we do however use the WWW interactive forms facility to support a set of client components that access the HOED server through a gateway. Both the overall aim and design of Hyper-G are more similar to ours.

7 Current Status and Plans for the Future

At the time of writing (March 1994), development of the HOED server is well advanced: although still at a prototype stage, it is now operational.The original conceptual data schema (developed in the Enhanced Entity Relationship model) was translated into a corresponding relational database schema, which was implemented, using a commercially available RDBMS. We have also developed prototype client components, including a simple query facility that enables end users to retrieve information about nodes, based on (boolean combinations of) simple search constraints regarding the characteristics of nodes, links or the participation of the former in the latter. Our plans for the immediate future focus on:

- As mentioned in section 5, we are working on different applications that will make use of the HOED environment. The development of *basic computer science modules* started in September 1993. Experiments for *access to local resources*, in collaboration with the university of Bristol, started in the summer of 1993. We are also further developing more client components.
- We will integrate within the HOED server facilities for Computer Supported Cooperative Work (CSCW) and collaborative learning. In a first phase, we will provide an annotation facility. Lecturers can use this facility to collaborate on course material and make their contributions accessible for lecturers and students in other institutions. Students will also be able to annotate their courses, e.g. as an aid while working through the material.
- The HOED server currently supports the basic hypermedia data model (links, nodes and webs). We are investigating how more advanced *data models* [CACM 1994] can be supported on top of the current model. In fact, modeling hypermedia data is not a trivial matter, because of the complex structure and multimedia content of hypermedia objects. This is therefore an appropriate case study for different data modeling paradigms. We have not elaborated on this issue here, but this is one of the reasons why we originally became interested in the HOED project.

8 Conclusion

Distributed, large scale multi-user hypermedia environments provide many exciting challenges in the area of data modeling, user interface development, communication protocols, information retrieval, etc. We are often excited about the *rapid developments* in these fields and the new applications they lead to. The opportunity to experiment with novel ways to deliver education is much more than just a practical outcome of these technical developments. The prospect this offers for innovation of the *educational practice* is quite a powerful stimulus for continuing our efforts.

References

[CACM 1994] Special issue on hypermedia. *Communications of the ACM*, 37(2), February 1994.

[Duval & Olivié, 1992] Towards the integration of a query mechanism and navigation for retrieval of data on multimedia documents. *ACM SIGIR FORUM*, 26(2):8–25, Fall 1992.

[Duval & Olivié, 1993] A distributed management system for multimedia and hypermedia educational resources. *Proceedings of ED-MEDIA 93*, pages 161–168. AACE.

[Duval, 1992] An 'intelligent' database system of audio-visual material. *Proceedings of the DELTA Conference on Research and Development*, pages 231–241. Kluwer Academic Publishers

[Engelbart, 1990] Knowledge-domain interoperability and an open hyperdocument system. In *CSCW 90 Proceedings*, pages 143–156. ACM, October 1990.

[Kappe, Maurer & Scherbakov, 1993] Hyper-G – a universal hypermedia system. *Journal of Educational Multimedia and Hypermedia*, 2(1):39–66, 1993.

[Nelson, 1993] *Literary Machines*. Mindful Press, 1993.

[Olivié & Duval, 1993] Basic computer science modules. In *CBLIS 93, International Conference on Computer Based Learning in Science*, pages 20–28, 1993.

[Obraczka, Danzig, & Li, 1993] Internet resource discovery services. *IEEE Computer*, 26(9):8–22, September 1993.

[Ousterhout, 1990] An X11 toolkit based on the tcl language. In *Proceedings of the 1991 Winter Usenix Conference*.

[Wiil, 1993] Experiences with hyperbase: A hypertext database supporting collaborative work. *SIGMOD Record*, 22(4):19–25, December 1993.

Distance Management Education: Process and Evaluation of Course Design, Manager Support and Media.

John Eldred,
Kingston Business School,
Kingston University, Kingston-Upon-Thames,
Surrey KT2 7LB, United Kingdom.
Email: BS_S426@KINGSTON.AC.UK

This paper examines some of the key issues arising from the delivery of a management development course by distance education using traditional, multimedia and computer-based approaches. The research explores some of the critical decision that need to be taken by educationalists involved in developing distance education, computer-based training, multimedia and hypermedia. The discussion focuses on the particular challenge of course development using these media where the course is fully or part distance or open learning in its form or method of delivery. The paper examines three key issue: What are the educational design issues that need to be addressed when developing distance education packages for the junior and middle manager ? How can we improve the way in which we use computers, electronic mail and networks for communication and student support ? What are the issues associated with the integration of multimedia and hypermedia systems into distance education for the middle manager ?. The paper concludes by critically examining specific futures that are advanced by different groups in education and training professionals.

Introduction

The development and effective delivery of Distance Management Education constitutes a significant challenge to traditional approaches to course development and course design. The parameters of the problems are defined by the complex and diverse character of the material being delivered, the contrasting and varying motivations and experience of the participating managers and the general difficulties involved in ensuring appropriate support to students who are physically dispersed from their peers, the course administration and teaching faculty.

A simple pedagogical strategy for meeting the needs of all the participants in this relationship can be elusive. The central task is to identify the teaching and learning design that can satisfy the needs of the managers who are recruited to our courses while ensuring the relevance of the methods of delivery and academic credibility. The rapidly evolving technologies of Multimedia and Hypermedia would seem to have a natural home within the established knowledge base that has developed in Distance Education. While the cliché of "the Media being the message" may seem familiar it is important that we analyse and evaluate the relevance of different media to the needs of our students. The process needs to consider the wider issues associated with the suitability of the media to the material being covered, the experience and motivations of the students, the approach and attitudes of the relevant Faculty and the learning support.

Development and Design of Open/Distance learning delivery systems

Open learning management development courses have an implicit assumption of being outcome oriented. The nature and purpose of the course is to change the managers ability and behaviour. In this sense the form and function of the course is strongly linked to the philosophy of competence and management development. The design of management development courses has changed dramatically as a result of a revolution in our expectation of what we hope to achieve in changing the attitudes, behaviour and performance of managers. The issue of how we define our educational objectives in has been the subject of much debate and criticism

(Macdonald 1983; Rowntree, 1987, 1992; Marshall, 1991). The central issue for open learning management courses is that our objectives should reflect the development perspective of the individual within a framework of independence and autonomy. Boot and Hodgson (1987) provide a clear and relevant distinction between two models of learning. They identify the traditional model as being characterised by learning being a process of dissemination. The contemporary alternative to this view is of a model of learning exemplified by development. A number of authors have supported this distinction and supported the development model . Rowntree,(1992) attempts to distinguish between to two distinct approaches. The contrast between the two approaches is presented in the table below(Table 1).

Table 1

Attributes and Approaches of the Dissemination and Development models of Management Education

Dissemination	Development
Focus on the acquisition of facts and procedures	Focus on growth and empowerment of individuals
Product and commodity oriented	Psychological/emotional process oriented
Utility and consumption focus	Needs and interests focus
Primarily teacher passes knowledge to learner	Two way process between teacher and learner
Munication dominant	Co-mmunication dominant
Abstract and theory	Concrete and experiential
Technical, knowledge and task	Soft skills, and openness

The implication of adopting a developmental model for the design of open learning management development course is that existing course components need to be revised and restructured. Furthermore new approaches to management education and learning need to be adopted in order to address this new set of development objectives. This new approach also involves reconsidering our existing model of the learning process (Burgoyne & Stewart, 1991). The adoption of a developmental approach in open learning managment development is actively linked with a number of major themes and theories now gaining acceptance in management education. Mumford identifies these themes as "action learning"; "the learning organisation" and "the competency approach".(Mumford, 1993).

The new approach to open learning management development involves the major exercise of restructuring all the components of existing courses. This process and the contrast between the old and new approach to managment development can be demonstrated by a comparison between the old and new versions of one of the open learning management development programmes offered at Kingston University. Kingston Business School has delivered an Open Learning Diploma in Management Studies for nearly 10 years. The Diploma course is designed for junior and middle managers and is studied over a period of two years. The initial course recruited 100-150 managers each year. The philosophy annd structure of this programme was redesigned and rewritten three years ago to reflect the adoption of a competence approach to management education and a developmental philosophy. During this process the author was involved in developing, evaluating and delivering more than 20 different management courses requiring in excess of 1000 hours of study. This portfolio embraces a comprehensive set of subject materials with a major emphasis on computer based and work related competence based assessment.

The new competence based open learning management development programme has clear benefits for the participating managers and the sponsoring organisation. The competence approach integrates the development process into the wider work environment of the individual. Work related assignments emphasise the personal needs of the individual and the improvement of performance. The flexibility of the competence approach using open learning study materials means that managers can concentrate on the development of a competence set that is relevant to the individual. The study materials, residential sessions, network groups are actively developed by the teachers and the learners in a partnership. The differences between the old and new approaches is presented in Table 2.

Table 2
Comparison of old and new designs for Open Learning Management Development Courses

	Old Course Structure & Approach	New Course Structure & Approach
Philosophy	Manager expected to learn the Theory, knowledge and techniques of management as an academic subject.	Manager develops and demonstrates competence in the application of skills and tools of management.
Objectives	Dissemination	Development
Participant Managers	Emphasis upon academic experience and qualifications for entry	Emphasis on empowerment and growth within the context of current and future responsibility. Importance of "openness" in terms of accessibility and flexibility.
Study Materials	Structure associated with text book but with exercises and model answers. May involve "commentary" upon recognised text book. The materials tend to be subject focussed with little overlap between subject areas. Developed and written on a three/five years development cycle and printed using traditional techniques.	High level of involvement and interaction. Work related activities and exercises. Skills and techniques form a matrix across different subject areas. The visual design and layout is an important issue to improve understand and motivation. The materials are updated and amended in a continuous process and in response to learner needs. Use just-in-time production .
Assessment Methods	Examination and essay type assignments.	Work related assignments or case study. Wide variety of formats from executive reports to individual presentations. Philosophy of assessment being potentially applicable to the work environment. The central purpose of the assessment is to provide feedback and advice as well as performance evaluation.
Residential Workshops	One week at the end of the course. Main themes revision and problem solving. Teacher lead revision sessions.	Weekend sessions phased throughout the programme. Designed to maintain regular contact and support. Wide range of activities workshops, group work, case study presentation. Flexible design to respond to learning needs of different groups.
Learning Support	Telephone help lines to administration and to the specific subject tutors.	Network study groups of managers providing mutual support. Peripatetic tutors visiting network groups. Email contact to course team and plans for video-conferencing.
Outcomes	Management qualification.	Qualification but emphasis on continuing development and learning.

The course extends the general philosophy of Open Learning into the design and development process, the key attributes of accessibility, flexibility and learner control remain the foundations of the course (Paul, 1990). A second measure of success is the very high completion rate within the set duration of the course. The course has the flexibility to enable managers to complete the programme over a period of five or more years. Even with this flexible provision typically 90% of the managers complete the course in the initial period of registration, the remainder will continue on the programme over a longer period of time. One of the major reasons for the success of this programme is the contribution of the support systems that have been developed for the course. These have had a profound effect on the experience and effectiveness of the programme.

Technologies and Systems for Supporting Managers

There has been a vigorous and active debate in management development circles focused on the selection and adoption of technology that can contribute to the effective and efficient management development course. The mail, .elephone, fax, and in recent years, email networks and video conferencing have all been utilised in the design of management development courses to provide an effective communications and support system for the manager and the course faculty. The philosophy and function of the support system is central to its form and effectiveness. In earlier designs the support system typically depended upon providing a substitute for the face to face tutorial . The hegemony of the traditional tutor system exemplified by the University tutorial modelled around the Oxford and Cambridge University was a fail-safe and crisis management approach to student learning and support. The telephone help line occupies a central position in most open learning management development course. The help line should be a mechanism for recognising and responding to the individual learning needs of the managers on the course. The problem with this model is that in practice the help line acts as a compensation for inadequately designed study materials and/or a failure in the provision of adequate formal support structures within the overall course design.

Research involving a qualitative evaluation of the experience of managers following a course with support from telephone help lines identified a range of key problems with this approach. The problems fall into three categories educational, contextual and logistical. Educationally the help line provides a highly constrained medium for the support of managers. Tutoring and supporting learners in complex, technical and in some cases abstract concepts is not easily performed on the telephone. While individual ability and experience will engender different support needs the study materials should be structured and designed to support the full range of experience and ability. In many cases the help line was used as an ad hoc mechanism for students to compensate for weaknesses in the materials or administration.

The second problem associated with help lines was the context of the support. The help line was seen by the anxious or less able student as the means for providing a compromise solution to replace the intensity of support offered by the face to face contact of traditional learning systems. When real or imagined spectres presented themselves to the learners the response was not to evaluate the issues or ideas and reformulate the problem but to pick up the phone and seek outside support and explanation, a ghost buster mentality. This approach links into the learning styles models discussed earlier (Boot & Hodgson, 1987), where open learning was seen in terms of dissemination with a consequent emphasis upon the learning being oriented towards facts, consumption of information and a clear distinction between teacher and learner.

Finally the logistics of ensuring help line contact between the manager and the course team was regarded as the main problem with help lines. The telephone is an immediate but an ephemeral medium. However, the support required may not be available or exist. The failure to achieve adequate response can lead to frustration and due to unfulfilled expectation a negative attitude to the management course. It can be argued that the answer phone goes some way towards reducing this problem but does not remove the problem of students being able to access the right person at the time of contact. This scope for failing to meet the expectations of the manager is apparent and a major cause of concern.

This experience and research into help lines lead to an explicit rejection of this approach in the current design of the management development programme. However, individuals and corporate clients tend to expect the provision of help lines as being a traditional component of this type of open learning course. The development and implementation of appropriate support systems has been a key feature of recent approaches to open learning management development. The central issue related to support is to reduce the managers need for support and increase the quality of response when support is required.

The foundations of this approach is to empower the managers on their course to accept responsibility for their own and their colleagues development. A learning contract is forged between groups of managers into network study groups. These network groups are given explicit responsibility for managing the learning process. The networks are required to meet regularly and to set and establish an appropriate mutual support system. This network is also used in the academic agenda of the course to provide the learning and assessment vehicle for several areas of the competence programme e.g. team membership, leadership, conflict management. The networks have reoriented the participating managers view of the management course. The need to seek support from each other has changed the experience of the course. The managers are now establishing and setting their own development needs and demonstrating and using their competence to support others. This approach allows the management course to fulfil a central philosophy of open learning that it should be an approach that enables independence and autonomy.

A second aspect of this approach is that groups are encourage to provide a critical input into the study materials in a formal and structured process of communication between the teachers and the learners. A prerequisite for this approach is the technical ability provide a rapid revision to the materials. This means having access to a resource of academics, designers and editors who can implement revisions. Furthermore a just-in-time approach to production is needed where materials are revised in real time. Recent developments in printing technology where traditional large volume/scale print runs are replaced with electronic publishing systems producing small print runs at an economic level is central to this approach. Thus, study materials evolve in an upwards cycle of quality improvement. The traditional cycle of three to five year development cycles and revision of materials is not relevant in this approach. The materials are relevant to the evolving needs and requirements of the managers. The problem of ageing and relevance has been a particular issue for a number of open learning courses in academic areas that are changing rapidly e.g. information technology management. This approach means that the need for support is significantly reduced.

A central requirement and goal for effective manager support is that when it is required it is structured and of a high quality. Individual differences between managers and groups may result in the need for additional support beyond that provided within the regular quarterly residential study weekends. The solution to this problem is to provide peripatetic tutor support that can be requested to attend meeting of the network group. This approach can be resource intensive and to be effective requires a large scale academic resource to draw upon. It is important to distinguish between this learner demand orientated support system and the more traditional approach in open learning of regular tutor meetings that are programmed throughout the duration of the course. The peripatetic network tutor approach provides the psychological support needed by the managers in the early stages of the course while it does not undermine the independence and autonomy of the learners. The managers are still clearly responsible for the learning and development process. The managers must evaluate their needs and formalise the request for academic support.

The experience of using this approach for the last three years has demonstrated that the support function would appear to be more a psychological need than actual. During this period approximately 50 network groups have been formed on the management course (the approximation is related to the dynamic nature of the groups dividing and coalescing). Only two network groups have organised more than five tutor visits in one year. The remaining groups are evenly divided between groups that requested one or two meeting, and interestingly, groups who had no tutor contact. The extremely low level of utilisation of the tutor support came as a significant surprise to the author. One explanation is that awareness of the availability provided initial reassurance after which learning support was provided by the networks and the quarterly residential workshops. A further point was that the formality of organising a meeting forced the managers to consider the scale, scope and content of the support required, this process served to act as an opportunity to reflect on the perceived learning problem and in many cases resulted in the groups reconciling their difficulties. A final issue was the possibility that much of the activity generated by students using help lines is more to do with a psychological sense of isolation than problems that require academic or administrative support. The network groups deliver the psychological support thus removing the need for faculty involvement in managers support.

The current planning process for the provision of quality manager support systems is directed towards the implementation of two communication technologies to improve the administrative and academic support provided to managers. The evaluation and provision of electronic mail and teleconferencing into the learning support systems of the management programme is being planned and implemented. Electronic mail (Email) has been available to the corporate and academic communities for many years. Indeed, open learning courses have established an extended track record of learner support using email. The main factor inhibiting the wider adoption of this technology by management development courses has been the complexity of the communications protocols, the need to have some familiarity with operating systems running on different platforms and the relative crudity of the communications networks. In recent years the technology has become simpler, more manager friendly and faster. The strength of email is its administrative formality and structured format. Email has a number of advantages over the telephone as a technology for supporting open learning managers. Communications are date stamped, documented and be easily filed/stored. The main strength of email is that it waits for a response or action, advises the sender of when the message has been read and allows easy and rapid reply.

The implications of the wider availability of teleconferencing or video-conferencing for open learning management is profound. The strength of the visual image as device for supporting learning , explanation and motivation is apparent if on examines some of the leading exponents of open learning education. The development of television and video based courseware has been innovative and a major source of innovation. Teleconferencing offers the promise of providing the benefits of flexible and interactive tutorial support at a distance and at a lower cost. The traditional use of closed or open circuit television to provide lecture sessions to widely dispersed groups of people has severe limitations as a medium for open learning. Video conferencing opens up access to high quality tutorial support, while still retaining the autonomy of the learners who can decide whether to participate or not. The current proposal is to proceed with the implementation and evaluation of a pilot project involving video-conferencing tutorial support of management development courses in the near future in association with British Telcommunications Plc. The contribution of the technological sophistication of ISDN (Integrated Digital Services Network) on video-conferencing and the accessibility of information for open learning is arguable the most important issue facing distance education initiatives as we approach the millennium.

Educational Media and Management development

The final theme of this paper is to evaluate the relevance and contribution of media selection upon the effectiveness of the open learning study materials. The choice of media is becoming one of the key decisions facing academics who are developing courseware. One guiding principle is that the process of selection is that the media selection should ensure that different stages of the learning process are enhanced and complemented by the use of the appropriate media. In recent years there has been a a significant increase in the available technologies that can be used in open learning courseware. The foundations of open learning courseware is in the printed media. The development and design of open learning study packages has become an established discipline with its own accepted philosophy and methodologies. The printed media has been complemented by a long tradition of using film and video materials to support the printed material. In recent years alternative technologies have offered a new functionality to the learning and development media. These new technologies have there origins in computing and multimedia systems and include hypertext, computer based training (CBT), interactive video, etc.

The challenge for these technologies is to demonstrate superior performance to the traditional printed media. In the context of developing appropriate systems for producing effective management development course a number systems have been piloted on the programmes at Kingston Business School. The attitudes and experience of managers who have used different media on management development programmes provides a valuable insight into the perceived utility and functionality of these technologies. One of the challenges facing those who are charged with developing open learning courses is identifying systems that are superior to the traditional forms of media. The printed media have specific strengths and a flexibility as a media for open learning study packages. Printed open learning materials are flexible in terms of the ability of the manager to study, select and interact with the courseware. Printed material is a low cost delivery system that does not require expensive hardware or software. Manager can use the material and study where and when they have time. Finally, printed material can be physically annotated and amended by the manager.

From a pedagogical perspective the key attribute of the printed study materials is that they represent a logical and linear structured process for aquiring the knowledge skills associated with a managment competence. The progression in management development is more difficult to manage than in other equivalent areas of open learning. The participants have a highly developed cognitive map of the structure and relationship between entities in the subject area. Each individual needs to select and sort from the available material there own specific pathway through this landscape. This makes the process of producing simple, logical and consistent designs for the material difficult if not impossible. Some would argue that this aspect of the problem lends itself to being specifically appropriate for the use of hypertext media that gives individuals the ability navigate around the material. This seems superficially attractive but it has a number of challenges.

Initial exposure to the concepts can be most beneficial if structured and ordered. The partnership between teacher and learner provides this inherent benefit. The current importance of the coaching analogy in management development also reflects this approach. The availability of a guide speeds the process of explorations into the terra incogniti of management education. It is in the later stages of consolidation, reflection and application that our pathway through the knowledge base becomes more eclectic and individual. It is during this phase of development that the flexibility offered by hypertext can offer a unique medium of support in the management development process.

The problem arises when one tries to establish some priority or selectivity in these linkages. The traditional printed media has incredible flexibility and utility and it can support a wide variety of styles, images and materials. While it has obvious limits to the range of media it can support it can be scanned and searched very flexibly, thus from the point of view of the learner it is accessible. The strength of hypertext and similar media for management education is the scale and scope of the information they offer and the ability for learners to manage the process of exploration. This is particularly valuable when consolidating the learning or developing new chains of association.

The important issue for the effective development of hypertext materials is that links and chains should be established by the learners, not by a reproduction of the teachers or coaches mental map. The existing media of print, video and film are perfectly capable of providing a consistent and effective representation of the initial or preferred path to explanation and understanding. Hypertext systems can complement this by providing the optimal vehicle for consolidating, developing and exploring the knowledge base. Given this role it is critical that the chains and linkages between element of the knowledge base should be defined by the learners not the teachers. This means recording how learners navigate around the system and recording and documenting positive search or exploration patterns. The building process is equivalent to the learning behaviour of neural networks where via an iterative process configuration to the nodes and links in the network are developed from the training of the neural network.

Conclusions

The design of effective Open learning management development programmes is very difficult. Providing courses that can meet the needs of participants, corporate clients and academic institutions is in many senses a hard problem to reconcile. In relation to this debate three issues need to be addressed:

- The philosophical foundations of Open Learning need to be integrated into the structure and content of the course design and culture.
- Open learning management development courses need to have effective learning support networks that are student centred in their focus. The support should be directed towards the goals of openness, flexibility and learner control. The model of support proposed in this paper attempts to meet this specification.
- There is an important role for multi-media and hypertext sytems in open learning management development programmes. It is important that we analyse the pedagogical role and function of these media and establish at what stage in the learning and development process they can provide the maximum contribution to the needs of the manager.

References

Boot, R.,& Hodgson, V., (1987). "Open learning: meaning and experience", Chapter 1 in V. Hodgson,S. Mann,R.Snell (eds) (1987) *Beyond Distance Teaching-Towards Open Learning*, SRHE & Open University Press, Buckingham

Burgoyne, J. & Stewart, R. (1991). "Teaching and learning methods in management development", in *Personnel Review* Vol 20, No 3 pp27-33

Macdonald, B. (1977) "An educational evaluation of NDPCAL" in *British Journal of Educational Technology*, Vol 8, No 3

Marshall, K. 1991) " NVQs: an assessment of the 'outcomes' approach to education and training" in *Journal of Further and Higher Education*, Vol 15, No 3, pp56-64

Mumford, A., (1993). "How managers can become developers" in *Personnel Management* Vol 25, No 6 pp42-45

Rowentree, D., (1987). *Assessing students: How shall we know them?*, Kogan Page, London

Rowentree, D., (1992). *Exploring Open and Distance Learning*, Kogan Page, London

Paul, R., (1990), *Open Learning and Open Managment*, Kogan page, London.

Electronic Classroom®:Features, Users and Evaluation Studies

A.ELLIS AND R.S.DEBRECENY
Faculty of Education, Work and Training and
Faculty of Business and Computing,
Southern Cross University,
P.O. Box 157 Lismore NSW 2480 Australia
aellis@alsvid.scu.edu.au & rdebrece@alsvid.scu.edu.au

Abstract: Electronic Classroom is an Australian developed software product(1) that provides a shared screen facility between two to six Apple Macintosh™ computers using standard telephone lines and modems. It provides an easy to learn icon based "paint" interface with point and click switching of control between sites. Used in conjunction with an additional voice line (loud speaking telephone) the software provides and easy to use, robust and relatively low cost audiographic teleconferencing system.

Electronic Classroom is already in use in distance education programs in every Australian state and mainland territory. Currently over 500 sites are being used to teach dozens of different subjects at all levels from primary school to university.

Evaluation studies have found that students using Electronic Classroom often felt involved in the learning process more than might be the case with "regular" classroom methods. They were able to clearly differentiate between different teachers and their respective teaching styles and preferred teachers who facilitated discussion rather than those who delivered a one-way address. For many Australian students the virtual classroom is already a reality.

History of Development

In Australia in 1987 the Victorian Ministry of Education pioneered the notion of linking clusters of remote schools via phone(voice), fax and computer. InterMac(2), a program produced by another Australian developer, was successfully used to prove the viability of teaching lessons using a shared computer screen. In late 1988 Robert Crago of Revelation Computing Pty Ltd, at that time the supplier of Timbuktu® and Timbuktu Remote®(3), recognised the need for a faster and simpler program that would provide a shared screen environment analogous to the classroom whiteboard. By early 1989 a prototype of Electronic Classroom had been built. Version 1.0 was commercially released in February 1990. Two years later and after some 14 minor upgrades, version 2.0 was released in July 1993. In September 1992 Electronic Classroom won a Merit award from the Australian Society for Educational Technology. The citation read "Electronic Classroom is the most effective tool for linking several remove classrooms under the control of a central tutor. It facilitates learning at a distance and encourages collaborative contact between participants. It offers interactive opportunities that would otherwise be denied to isolated students...." Further details of the history and development of Electronic Classroom are provided in Crago (1992) and the regular newsletter, Electronic Classroom Update, issued by Revelation Computing.

Design Characteristics

Much of the appeal and success of Electronic Classroom is derived from its ease of use in terms of both establishing and maintaining contact between remote sites and in conducting interactive lessons using pre-prepared screens.

(1) Electronic Classroom® was developed by Robert Crago of Revelation Computing Pty Ltd, P.O. Box 356, Zillmere, 4034, Queensland, Phone +61 7 2638891 Fax +61 7 2638871, AUST0141@applelink.apple.com
(2) InterMac is no longer under development.
(3) Timbuktu® and Timbuktu Remote® are designed as a screen sharing product for existing Macintosh™ applications and are best suited to local area network use.

Electronic Classroom adopts a teacher-student-whiteboard analogy. The site that initiates the dial up links (the teacher) handles all connection procedures. Student sites only require the modem to be turned on and the application to be opened. As part of the design philosophy, which stresses simplicity of use, a set of "paint" rather than "draw" tools are provided (Figure 1). The tool palette, much like an early version of MacPaint®, contains the usual display elements of text, lines, rectangles, ellipses plus a rubber. Fills, line shading and colour options are available under the *Paint* menu. The result is a user friendly interface.

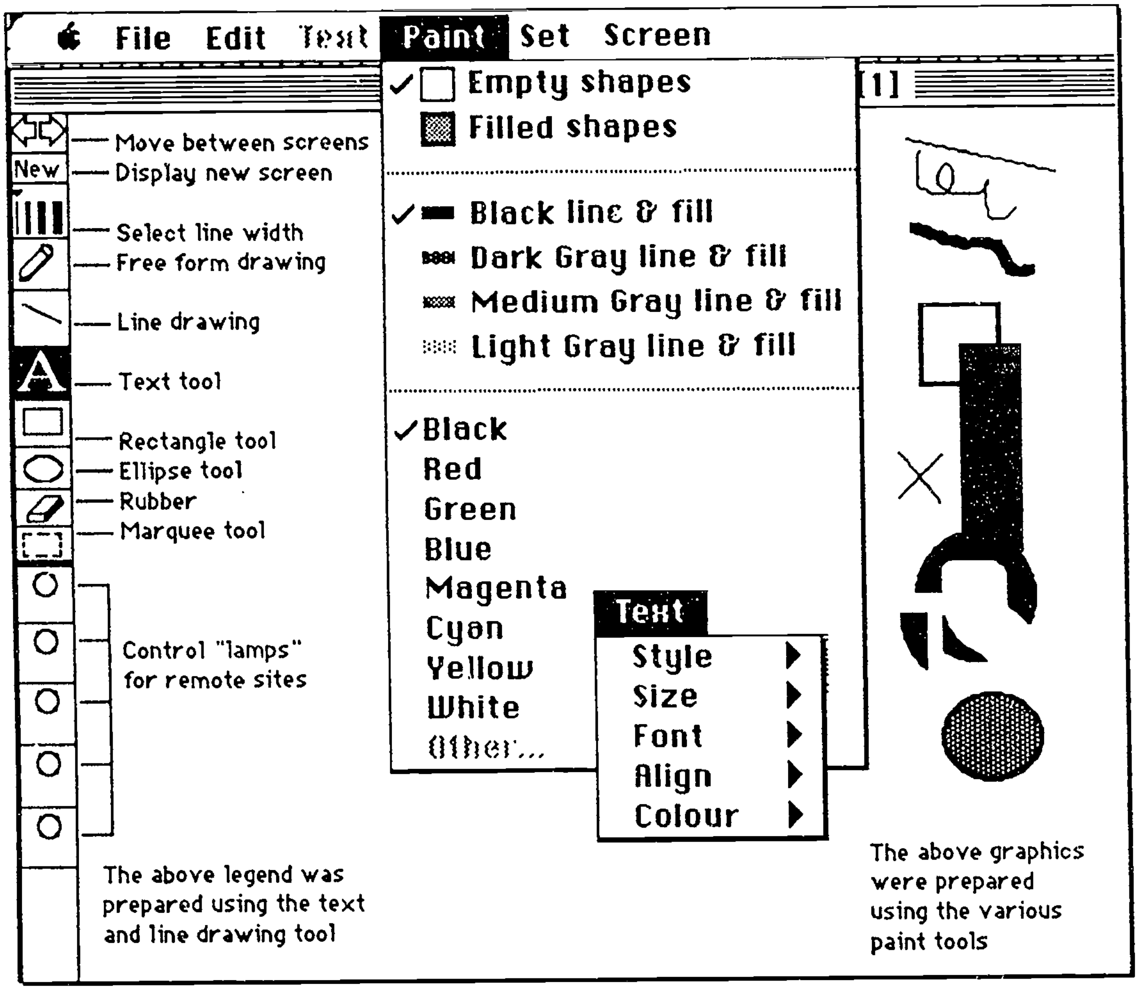


Figure 1 Top left hand corner of standard Electronic Classroom screen showing tool palette, *Paint* menu and superimposed *Text* menu.

Control of the shared screen is determined from the teacher's site with an audio signal (buzz) and flashing location lamp being used by a student site to indicate that they would like to take control of the screen - the electronic equivalent of a student raising their hand in the normal classroom setting.

Linking of three locations (the teacher plus two student sites) can be achieved by using both the modem and printer ports on one computer. To connect to an additional three sites, termed multipoint communication, it is necessary to install a Nubus card (4) with additional serial ports in one computer. The software will recognise any additional serial ports that are plugged in and will add their names to the end of the *Set* menu. The computer fitted with the multipoint card may be used as one of the lesson sites or simply set to act as a *server* with other sites dialling and connecting to it or *vice versa*. Teacher control may be initiated from any site.

At individual sites the number of students and the type of interaction expected may require either screen projection facilities, multiple screens and/or the "daisy chaining" of several keyboards. Likewise the hardware needed to handle the audio line needs to be appropriate to the size of the group at the individual sites.

(4) An example of a suitable board is the Hurder®-HDS/HQS available for Creative Solutions Inc, 4701 Randolph Rd, Suite 12, Rockville, MD 20852 USA, phone 301-984-0262

The Features of Electronic Classroom V2

A major feature of all versions of Electronic Classroom is that they will run on any Apple Macintosh™ computer from the humble Mac Plus with its 9 inch monochrome screen through to any of the high end colour machines with screen sizes up to A3 size. The program is designed handle data transfer speeds at anything from 300 to 19,200 baud. However a connection at 2,400 baud is required for reasonable data transfer speed and 9,600 baud is preferred. The former speed is almost universally achievable on the current Australian PSTN and the latter frequently achievable. For training purposes computers can be linked directly using standard printer cables. All features of the program, except the dial up procedures, are available in this direct connection mode.

For any given link up all the computers involved must have the same setting for the parameters of screen size and baud rate and preferably screen colour (monochrome or colour). As any lesson is likely to use more than one screen of information and as multiple screens are held in RAM, knowing the memory available to each computer in the link up is crucial to session planning. The *Configurations* item under the *Set* menu gives details on each computer in any link up (Figure 2). In terms of cost and performance the ideal computer for the school environment is probably a Macintosh™ LC with 8MB of RAM, 14 inch colour screen and a 9,600 baud modem.

The features of Electronic Classroom were progressively improved with each minor upgrade of version 1. With the release of version 2 significant new features were added to the program and these included: remote CD-ROM access, Quicktime™ and Photo CD support, improved screen sizing, the ability to inspect a remote machine's physical configuration, software version number and memory size, the ability to extract text for PICT files and to print one or two screens per page. Additional features are already planned for future upgrades based on user feedback and hardware developments.

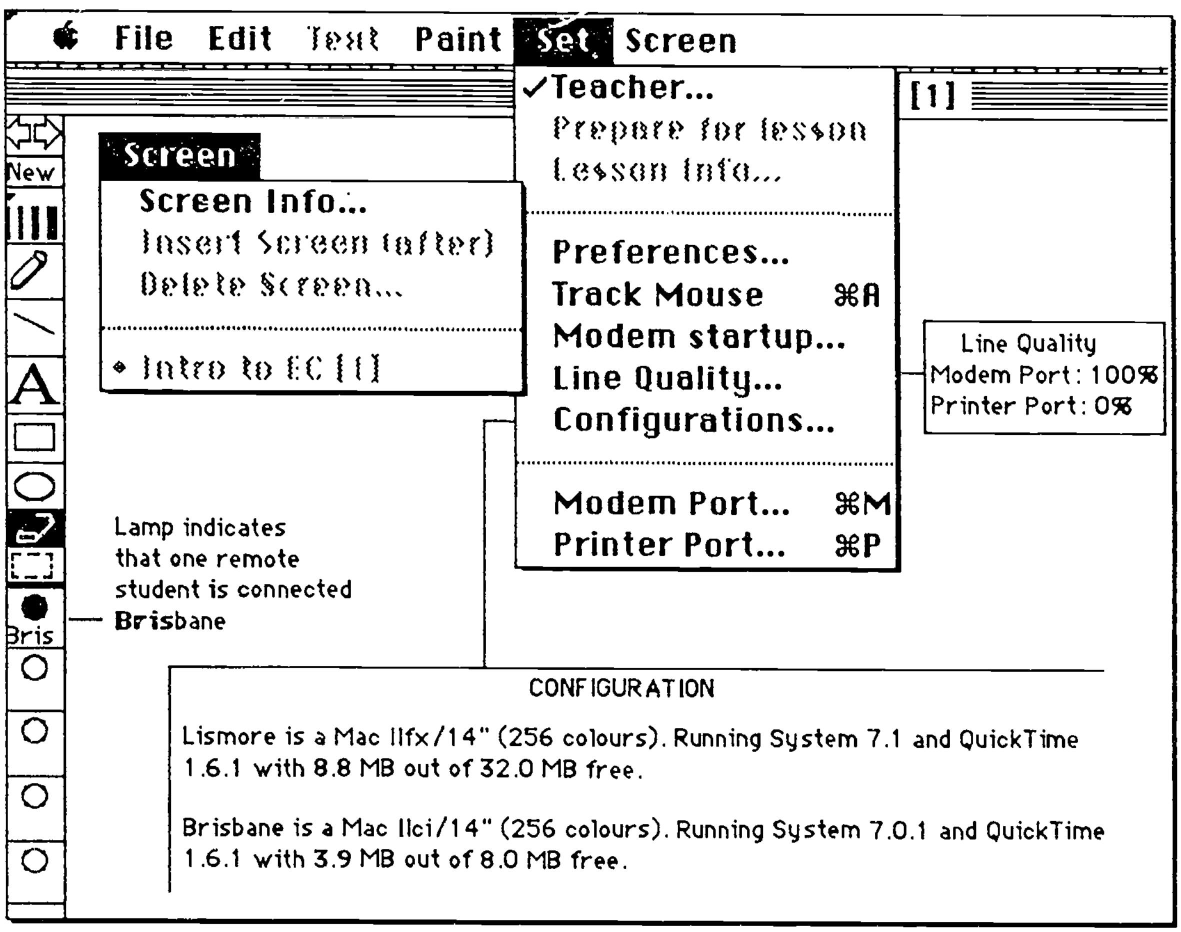


Figure 2 Shows the left hand corner of the screen after the Teacher has connected with a site named Brisbane. The *Set* menu and superimposed *Screen* menu options are shown as well as the information available under the *Configurations* and *Line Quality* items. With two computers connected by their modem ports and a printer cable the line quality is 100%.

Establishing a lesson

If you are participating in a lesson at a student site all you need to do is switch on the computer mand modem and start the application by double clicking on its icon. It will then check the state of the your serial ports for conformity with the configuration last used. For example, if the previous lesson was taught with a modem connected to the modem port, then the application will display an error if a modem cannot now be detected on the modem port. This alerts the user to possible equipment problems or perhaps the fact that the modem has simply not been turned on. With the application started and the equipment checked each site will be viewing a blank *Untitled* screen with a greyed out tool palette and at least one matching light grey status lamp (labelled M•P or P•P) visible.

A lesson can be initiated from any site by activating the *Teacher* item under the *Set* menu (Figure 1) . This requires the entry of a password, the name of the the teacher and the name of the lesson. The latter two items are transmitted to the other sites as the connections are established. The *status lamps* confirm a connection has been successfully established and indicate the first few letters of the remote site eg Bris for Brisbane in Figure 2.

Maintaining connections

As noisy or crossed telephone lines are still common in rural Australia the communications protocols built into Electronic Classroom detect such situations and continue to attempt of send information until it gets through. The *status lamps* and various dialogue boxes keep sites informed of the progress of information transfers.

Catering for different teaching and learning styles

Electronic Classroom can be used to cater for a range of teaching and learning styles. Firstly, it can be used to create all material during the actual lesson. Input can be via the keyboard, mouse or by cutting and pasting from the scrapbook or other files. New screens are created as required and both teacher and student can have input. Secondly, a number of screens can be prepared before the lesson and stored as one or more *Lesson Notes* files. Apart for the special Electronic Classroom *Lesson Note* file format, it is also possible to incorporate material from other files eg MacPaint® or PICT via the *Open* options under the *File* menu (Figure 3).

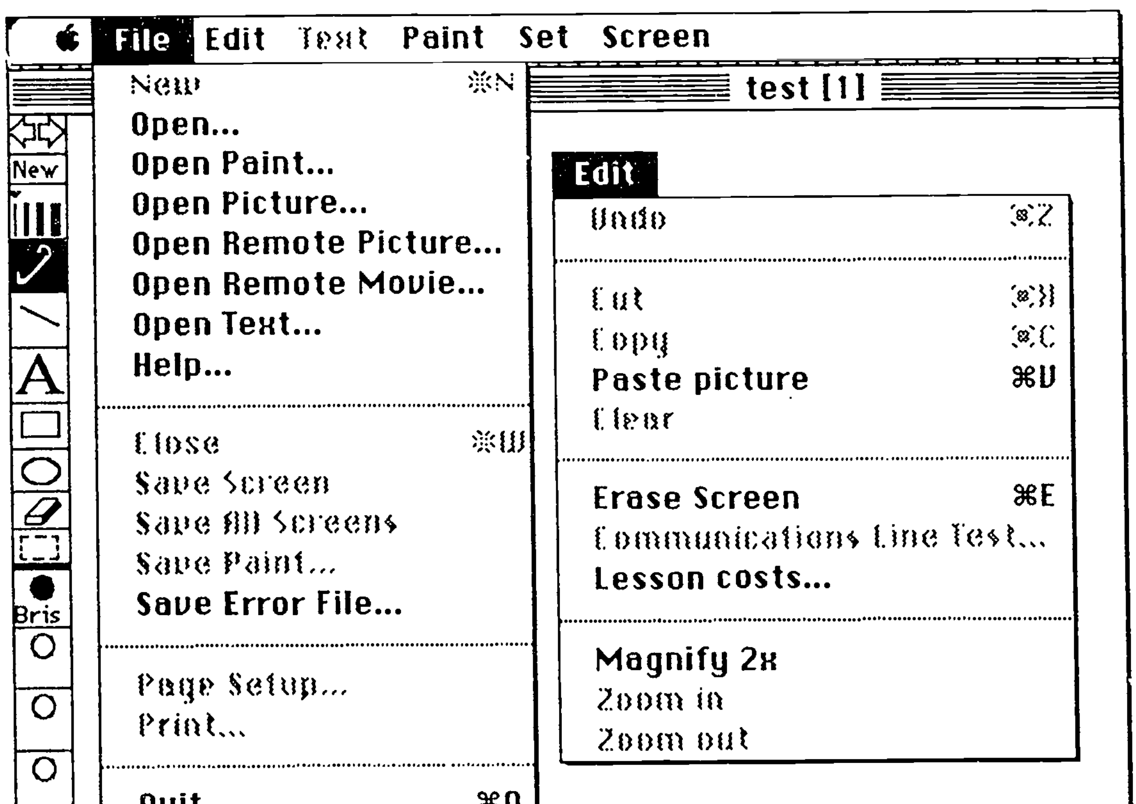


Figure 3 Top left hand corner of standard screen showing *File* menu and superimposed *Edit* menu. Note the wide range of Open ... commands including the *Open Remote...* facilities.

Who uses Electronic Classroom?

Electronic Classroom is already in use in distance education programs in all Australian States and mainland Territories. For many Australian students the 'Virtual Classroom' is already a reality. A selection of programs that use Electronic Classroom are briefly described below.

Primary schools

Some of the earliest and most successful projects are based in Victoria where the State Ministry of Education has established a support unit. Bendigo, a provincial city north of the Melbourne, is the hub of a distance education network that extends to hundreds of schools throughout the State These schools work in clusters, for example, the King Valley Cluster of four small rural primary schools (children 5 to 12 years old) use audiographic sessions as part of their Italian language teaching program. Indeed the teaching of Languages Other Than English (LOTE) has received a major boost from the introduction of audiographics technology into classrooms. Workshops attracting over 40 teachers have been held in Bendigo to develop computer-based material for the teaching of Italian, French, German, Japanese and Chinese.

In Queensland a network of 120 primary schools are involved in a LOTE program where secondary language teachers take lessons with clusters of up to five primary schools simultaneously. In this example the technology brings specialist teaching into the primary classroom setting.

Secondary schools

In Queensland a major 'Telelearning' project has involved the linking 23 secondary schools to teach a range of subjects to students aged 12 to 17 years. An important feature of this project is that the lessons delivered using Electronic Classroom are not undertaken in addition to pre-packaged distance education courses but directly parallel the program being taught in traditional face-to-face classes. Teachers taking Telelearning lessons use their own classroom materials, which are only modified, where necessary, for electronic transmission. In metropolitian Brisbane, the captial of Queensland, links between secondary schools as little as 6 kilometres apart have allowed for expanded student subject choice.

In Tasmania students studying Japanese at Murray High School students link with a teacher at the Tasmanian School of Distance Education. In Northern NSW a group of Central schools clustered around Moree use Electronic Classroom as part of a program to extend student subject choice in senior years (17 and 18 year olds).

Technical Colleges and Universities

In Victorian and Western Australia, Technical and Further Education Colleges already provide return-to-study and adult literacy programs and are trialing the teaching of accredited subjects.

At Southern Cross University sections of undergraduate degree courses are delivered from the Lismore campus to students at Open Learning Access Centres in Tweed Heads, Grafton and Port Macquarie. Postgraduate students enrolled in training and development courses have been involved in trial interstate links to share information on the last developments in industry responses to Commonwealth Government initiatives such as the Training Guarantee Act. The University has also trialled the teaching of short courses to industry and community groups. One such course involved motor vehicle repair firms and brought together a number of owner operators for training in accounting and costing techniques.

At Deakin University in Victoria the Faculty of Primary Teacher Education has linked preservice primary teacher education students on their Burwood campus with children in classrooms in the Castelmaine area. In the pilot program student teachers prepared and delivered weekly Electronic Classroom lessons on Indonesian language. During later teaching practicums the students lived in Castlemaine and taught the same children in the traditional classroom setting. Several other teacher training institutions are now considering training in audiographic technologies as part of their preservice programs.

Subjects taught

Across these various education sectors the subjects already taught using Electronic Classroom include: Accounting, Adult Literacy, Agriculture, Australian Studies, Botany, Chemistry, Chinese Computer Science, Economics, English, French, Geography, German, Graphic Communication, History, Human Development and Society, Indonesian, Information Technology, Italian, Japanese, Legal Studies, Mathematics, Music, Physical Education, Physics, Politics, Photography, Psychology, Secretarial Studies, Shorthand, Small Business Management, Technology Studies, Training Methods......and this list is growing.

Evaluation studies

Gray and Grady (1993) report on the use of Electronic Classroom in twenty three high schools in the Queensland (Australia) education system. The authors undertook audiographics hookups with two schools and administered standard questions at the end of each semester. They found that a particular issue was the interrelationship of the development, largely by teachers in the field, to the bureaucratic structure of school administration. There was clearly an underestimation of the level of commitment to the program required by individual teachers and by administrative systems. They found that the methods used to train teachers in the use of the technology and the "theory" of distance education, by bringing in teachers for a week long course, was inappropriate. Training sessions should have been centred at the schools and been "hands on" from the outset.

The study found that students felt involved in the learning process more than might be the case with "regular" classroom methods, and that their capacity for self-learning and to express themselves was improved by the use of audiographics. At the same time, the "classroom" analogue and the scheduled class "meeting" ensured that the students did not require too great a leap from the reality of day to day school life.

Debreceny and Ellis (1994) report in detail on two case studies at the secondary and tertiary levels in programs administered in New South Wales, Australia. At the secondary level, the authors report on the Northern Border Senior Access (NBSA) program which is an initiative of the schools of the Northern Border region of the North-West of the state. It joins together the "central" schools of four remote townships to introduce a senior high school and are taught as a combined school and use a combination of local instruction, telematics and audiographics. The program particularly caters for students who for family financial reasons would otherwise be unable to undertake secondary studies as a boarder in a regional high school.

At the tertiary level Southern Cross University operates three Open Learning Access Centres (OLACs) in rural towns in Northern NSW. Units in the Bachelor of Business degree are offered in a variety of modes including the provision of traditional distance materials supplemented by OLAC visits and teletutorials through to units taught in "internal" mode via Electronic Classroom with the same materials, class exercises and assessment requirements as the "on-campus" students. Students found that the regular weekly contact via audiographics was preferable to the use of traditional distance materials. As adult learners they were clearly less comfortable in the use of the technology than their counterparts in the secondary system.

The findings of the case study, which was undertaken with a particular focus on the student, confirm many of the findings of Gray and O'Grady. Students were keenly aware that if it were not for audiographics, their access to Years 11 and 12 could be denied. They were able to clearly differentiate between the teaching styles of the different teachers in the programme. They had a marked preference for those teachers who did just not talk whilst a screen was on display on the computer but drew the students into the discussion. They found, however, that the teachers were clearly better prepared for their audiographics periods than they were for the regular classroom sessions.

The students made a number of complaints on aspects of the technology in use. They were particularly frustrated at the time it took to hook the different schools up at the start of the lessson. It could take as long as 10 to 15 minutes to get hooked up and transfer all the screens at the start of the class. This was seen as being wasted time and restricted the amount of time that was then available for the class session. There were also problems with modem lines dropping during the class in perhaps one class in four and the time it took to transfer screens of information.

In general students were happy with the use of the technology. Some students thought that Electronic Classroom was very worthwhile with one student saying that audiographics was the "whole foundation for Year 11 and 12". Other students found that ".. audiographics is just, you know, school. School is school.".

Acknowledgments
The authors have no financing, consulting or other relationship with Revelation Computing whom they wish to thank for providing demonstration copies of of Electronic Classroom® for use by delegates at ED-MEDIA '94.

References
Crago,R.(1992) Electronic Classroom - Australian Technology for Distance Education. In *Proceedings of the 2nd International Conference on Information Technology for Training and Education*, Brisbane: University of Queensland.
Crago,R. (1993) Electronic Classroom, Version 2, 62p, Brisbane:Revelation Computing.
Debreceny, R. S., & Ellis, A. (1994). "An Evaluation of an Audiographic Delivery Method: A Cross-sectoral Study". In *Proceedings of APITITE94*,. Brisbane, Qld: APITITE.
Gray,A., & O'Grady, G. (1993). "Telecommunications Pedagogy for Effective Teaching and Learning". In G. Davies & B. Samways (Eds.), *Teleteaching*, Amsterdam: Elsevier Science Publishers.

It's Not How Multi The Media, It's How The Media Is Used

R. FEIFER
United States International University, 10455 Pomerado Road, San Diego, CA 92131
E-Mail: feifer@ils.nwu.edu

L. ALLENDER
Institute for the Learning Sciences
Northwestern University, 1890 Maple Avenue, Evanston, IL 60202, USA
E-Mail: allender@ils.nwu.edu

ABSTRACT: Multimedia educational software is often a glitzy version of old technology. This is not a new phenomena, early educational software fell into the same trap. Developers depended on the sexiness of the computer to make page-turning software more compelling. Some educational software has become better as developers began to ask, "In what ways can the computer facilitate learning, that were not possible before?" One answer to this question is: provide a simulated environment for the learner to interact with.

For multimedia to have an impact on learning we must ask a similar question, "In what ways can multimedia facilitate learning, that were not possible before?" One answer is what we call Case-Based Learn-by-doing Environments (CaBLE). The computer provides a simulated environment that allows the student to learn a task by doing the task. Multimedia stories and information helps connect what the learner is doing in the simulated world with events in the real world.

THE DARK AGES OF EDUCATIONAL SOFTWARE

Traditional computer aided instruction often use the following paradigm: the learner reads some content and then is tested on his knowledge; if the learner fails the test, then the process is repeated.

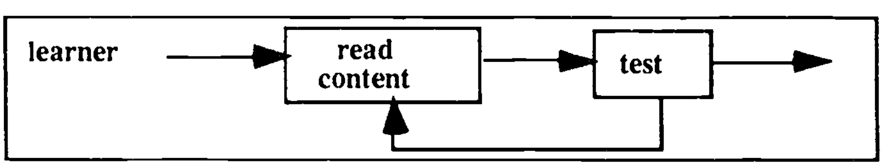


Figure 1. learner interaction with some instructional software

In this paradigm, the technology is used to evaluate the learner's test answers and to generate new lessons which may provide remedial information. Even given the best of implementations, this paradigm has the following problems:

1. the only motivation for reading the content is to pass the test;
2. the learner is exposed to the content often without an understanding of the context in which it would apply;
3. the approach emphasizes applying the right label to a concept, rather than the use of that concept in an appropriate situation.

A designer might attempt to augment this paradigm by delivering the initial content in a multimedia format, or presenting custom multimedia feedback for each wrong answer in the test. Although a multimedia version of the paradigm is more appealing, it will have the same basic flaws.

EXPLOITING COMPUTERS

Some designers have improved on this paradigm by utilizing the computer to create a simulated task environment for the student. Researchers have argued that the best way to learn how to do something is to try to do it and learn by your mistakes (Collins, Brown, & Newman, 1989; Laird, Rosenbloom, & Newell, 1986; Saxe, 1992; Schank & Jona, 1991; VanLehn, 1988). So instead of having learners simply read the content, learners take real life actions.

Simulated task environments broaden the types of training available. Actions in real life may be too expensive or dangerous for a student to attempt without the proper skills (i.e., flying a plane or disarming a bomb), but on a computer learners can actually learn by doing in these otherwise inaccessible task environments, (Govindaraj, 1988; Lesgold, Lajoie, Bunzo, & Eggan, 1991; Schauble, Glaser, Raghavan, & Reiner, 1991; Williams, Hollan, & Stevens, 1981).

Although simulations can make learning by doing an efficient instructional design, three factors limit the use of computer simulations in teaching:

1. good simulations are hard to build;
2. learners can flounder with just a simulation;
3. learners may not believe the simulation.

Building a good simulation in any but the most trivial domains is difficult. Whether you are simulating a physical system or a social system, allowing the user a wide range of actions and simulating the results of any combination of those actions requires a complete model of the domain.

Even if we could build a simulation with sufficient fidelity, it is difficult to learn from mistakes without some guidance (Kuhn, 1989; Schauble, et al., 1991) In particular, without guidance, learners who have flawed strategies may fail to successfully learn the skills targeted by the simulation. Without guidance, failure can only show you what not to do. It is often not obvious why you failed or what you should have done instead.

One solution to this floundering problem is the addition of a computer-based coach to watch over the learner's shoulder and advise (Burton & Brown, 1976; Goldstein, 1979). The coach, however, also requires a good model of the domain, plus a model of the learner. Even if we could model a learner's misunderstandings in a simulation, we still have the problem of generating a dialog to ameliorate those misunderstandings.

The last problem is that a student may not believe that a simulation is accurate. After all, even the best simulations are not a complete representation of reality. Multimedia technologies can link the simulation to real events.

EXPLOITING MULTIMEDIA TECHNOLOGY

We have used multimedia to provide this link to reality in a model we call Case-Based Learn-by-doing Environments (CaBLE). In a CaBLE tutor, instead of the computer generating text based instruction, good story tellers, experts in the domain, tell their stories on video tape. These stories are then indexed to the kinds of failure for which they are relevant. This enables the system to react to the learner's failures as a good teacher might, by recalling a real and personal story containing the principles to be learned from the failure.

These stories make teaching through simulations more practical in three ways:

1. the simulation need only provide a context and motivation for the story, the stories make up for any lack of depth or fidelity in the simulation;
2. it is easier to index failures than to model the learner sufficiently to provide intelligent coaching;
3. it is easier to show a video than to generate instruction, and more compelling to the learner;

The interaction with a CaBLE tutor is illustrated in Figure 2. The test from Figure 1 is replaced in a CaBLE tutor with the simulation of a real-world task. There is no test in the conventional sense of the word, successfully completing the task is evidence of mastery. In addition, the order from figure 1 is reversed. The task comes first. The learner is exposed to background material only later, whenever he/she wants or needs it.

If the learner does not know what to do, he/she can ask questions that are answered either by information declarative information or a story. If the learner attempts the task but experiences some failure, the tutor will present a story. The tutor chooses a story that will either help the learner explain the failure, or discover a strategy for avoiding the failure in the future, or both.

MAKING IT REAL

In a CaBLE tutor, multimedia technology strengthens the connection between what is being taught, and reality. If a learner makes a mistake in a conventional tutoring system, and the system merely tells him that he has made a mistake, it will have little impact. The learner may not even believe that it was truly a mistake. Maybe the computer did not understand what he did.

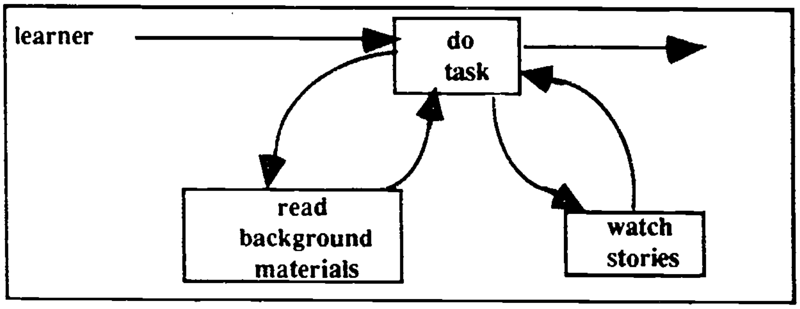


Figure 2. learner interaction with a CaBLE Tutor

If we wait for the mistake to lead to a failure within a simulated world, and then tell the learner that the mistake led to the failure, there is a better chance that the learner will see why it is a mistake. The learner will at least see why, in the tutor's opinion, it is a mistake. He may still, however, doubt that the linkage between the mistake and the failure would occur in the real world.

In a CaBLE tutor we go one step further. After the mistake has led to a failure in the simulated world, the system presents a video of a person describing a first person experience with a similar mistake, and the real-world failure that it led to. Although we could use text to tell the story, a video of a real expert provides the same information in a more credible format.

CABLE PARTS

A CaBLE tutor utilizes the following components: a task simulator, an interface, a library of failures which can occur in the domain, a library of learner mistakes, a learner state map, a network of declarative knowledge and a library of stories.

SIMULATION

The simulation of the task indicates the probable results of any learner action, if that action had occurred in the real world. The CaBLE tutor checks after each learner action to see if the mistake library recognizes that action as a mistake and to see if any resulting events are recognized by the failure library.

INTERFACE

The interface provides the learner with the means to take actions he/she might expect to be able to take in the actual task. In addition the interface communicates the results of the actions as determined by the simulation. The interface also contains generic question buttons that are interpreted by the current learner state to provide access to the knowledge network.

FAILURE LIBRARY

The failure library contains a list of events that would be considered failures in the domain (i.e., plane crashing or patient dying). Any time an event occurs in the simulation that matches one of these failure events the CaBLE tutor will check to see what mistakes have occurred that could have led to that failure.

MISTAKE LIBRARY

The mistake library contains a list of the common errors made in the domain. For each mistake there is a set of conditions that would allow the CaBLE tutor to recognize that the error occurred. In addition, each mistake is associated with a list of failures that could occur if the learner made the particular mistake. Each mistake is linked to appropriate stories and/or information in the knowledge network so that these can be offered to the learner after a failure.

LEARNER STATE MAP

The learner state map allows the CaBLE tutor to recognize the context in which the learner asks a question. This allows the tutor to provide more relevant information in response to the question. The learner's current location in the learner state map is updated when key interface actions occur and when key events occur in the simulation.

KNOWLEDGE NETWORK

The Knowledge Network contains the declarative knowledge the learner would have to have to be successful at the task. When the learner asks a question, the CaBLE tutor will bring them to an appropriate piece of information in the network.

STORY LIBRARY

The story library contains explanations, positive examples and negative examples for the domain being taught. When a failure occurs, the CaBLE tutor finds and tells a relevant story.

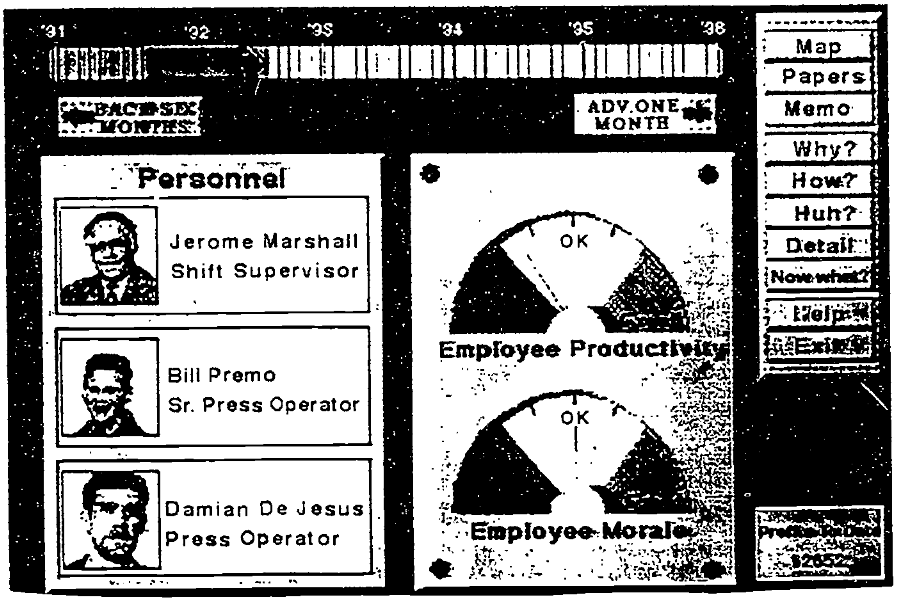


Figure 3: HeRMiT Opening Screen

AN EXAMPLE CABLE SYSTEM

HeRMiT is part of a business practices course designed to teach the principles of human resource management (HRM). The learner is asked to accomplish the following task (Bell & Feifer, 1992; Feifer & Hinrichs, 1992):

> You are the manager of the human resources department of the case company. You must make human resource decisions for the next four years, without destroying the company. To make your job easier, you will only make decisions for three employees. The computer will generalize your decisions to the rest of the staff.
>
> You can monitor the health of the company by watching the productivity and morale meters. If either of these meters goes into the red, you will be fired.

Behind this task is a simple simulation of employees and how their attitude, competence and performance interact to affect the overall success of a company. The learner must make appropriate long-term decisions for each of three employees, as well as respond to ad hoc situations as they arise (see figure 3).

The learner opens an employee's personnel file by clicking on their picture button on the main screen. The learner determines how the employee is doing by checking their static information, current attitude, competence or performance, paper trail, or history. They can also get more information as the result of initiating a formal evaluation or an informal "counseling" session. The learner takes action by changing the employee's position, salary, status or scheduling training. When the learner decides that they have taken all appropriate actions for a given month, they click on the "advance 1 month" button.

ASKING QUESTIONS

If the learner knows everything there is to know about HRM, he/she continues through the simulation rather quickly. If the learner lacks the basic principles, he/she can ask the tutor questions. The combination of the question button pressed and the learner's current context, permits the tutor to show an appropriate piece of information (Jona, Bell, & Birnbaum, 1991). If the information shown raises further questions, the learner can press a follow-up question button.

UNDERSTANDING FAILURE

When the learner makes mistakes he/she receives the same response he/she would receive in the real world: productivity goes down or employees become unhappy. When an employee quits or the company goes bankrupt, the tutor helps the learner understand what actions led to the failure and why. The learner is first shown a list of mistake types that are possible causes for the failure (Figure 4).

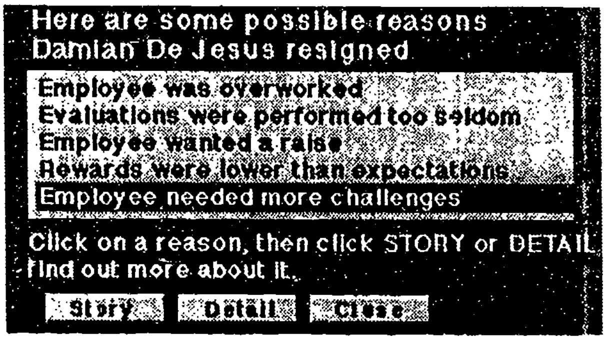


Figure 4. Failure Dialog Box

For each of these mistake types the learner can see:

- text information that explains why this type of mistake can be a cause of this failure;
- a video-taped story that provides a concrete example of this mistake type causing a similar failure.

SUMMARY

CaBLE tutors are designed to better exploit the potential of computers and multimedia technology.

The computer is exploited to provide a task that is challenging and interesting. We have found little need for any extrinsic motivation to induce a learner to sit down and attempt the task presented in HeRMiT. Further, once involved in the task, they will stay with it until they succeed.

But the above can also be said of most video games. In order to be successful in the task the student must know and/or learn whatever it is we want to teach. The tasks are designed so that it is impossible to take random actions and successfully achieve the goal. The task is the test.

Hypermedia allows the learner who cannot accomplish the goal with their existing knowledge, to learn new knowledge. They learn only on demand, and only in a context in which the learning will be meaningful.

When a learner takes an action that leads to a failure, multimedia allows the learner to see a real person telling a real example of that mistake and its consequences.

REFERENCES

Bell, B. L., & Feifer, R. G. (1992). Intelligent tutoring with dumb software. In C. Frasson, G. Gauthier, & G. I. McCalla (Eds.), *Intelligent Tutoring Systems* Berlin: Springer-Verlag.

Burton, R., & Brown, J. S. (1976). A tutoring and student modeling paradigm for gaming environments. ACM SIGCSE (Computer Science and Education) Bulletin, 8, 236-246.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (Eds.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum Associates.

Elliott, E. S., & Dweck, C. S. (1988). Goals: An approach to motivation and achievement. *Journal of Personality and Social Psychology*, 54(1), 5-12.

Feifer, R. G., & Hinrichs, T. R. (1992). Using stories to enhance and simplify computer simulations for teaching. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, . Indianapolis, Indiana: Lawrence Erlbaum Associates.

Feifer, R. G., & Soclof, M. S. (1991). Knowledge-based tutoring systems: Changing the focus from learner modelling to teaching. In *Proceedings of the International Conference on the Learning Sciences*, (pp. 151-157). Northwestern University, Evanston, IL: Association for the Advancement of Computing in Education.

Goldstein, I. P. (1979). The genetic graph: A representation for the evolution of procedural knowledge. *International Journal of Man-Machine Studies*, 11, 51-77.
Govindaraj, T. (1988). Intelligent computer aids for fault diagnosis training of expert operators of large dynamic systems. In J. Psotka, L. D. Massey', & S. A. Mutter (Eds.), *Intelligent tutoring systems: Lessons learned* (pp. 303-321). Hillsdale, NJ: Lawrence Erlbaum Associates Publishers.
Jona, M., Bell, B., & Birnbaum, L. (1991). Button theory: A taxonomy of student-teacher communication for interface design in computer-based learning environments. In *Proceedings of the Thirteenth Conference of the Cognitive Science Society*, . Chicago, IL.:
Kuhn, D. (1989). Children and adults as intuitive scientists. *Psychology Review*, 96(4), 674-689.
Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). *Universal subgoaling and chunking: The automatic generation and learning of goal hierarchies*. Hingham, MA: Kluwer Academic Publishers.
Lesgold, A., Lajoie, S., Bunzo, M., & Eggan, G. (1991). SHERLOCK: A coached practice environment for an electronics troubleshooting job. In J. H. Larkin & R. W. Chabay (Eds.), *Computer assisted instruction and intelligent tutoring systems: Shared goals and complementary approaches* (pp. 201 - 238). Hillsdale, NJ: Erlbaum.
Norman, D. A. (1986). Cognitive engineering. In D. A. Norman & S. W. Draper (Eds.), *User Centered System Design* (pp. 31-61). Hillsdale, NJ: Lawrence Erlbaum Associates.
Saxe, G. B. (1992). Studying children's learning in context: Problems and prospects. The *Journal of the Learning Sciences*, 2(2), 215-234.
Schank, R. C. (1991). Case-based teaching: Four experiences in educational software design (Technical Report No. 7). The Institute for the Learning Sciences, Northwestern University, Evanston IL.
Schank, R. C., & Jona, M. Y. (1991). Empowering the student: New perspectives on the design of teaching systems. The *Journal of the Learning Sciences*, 1(1), 7-35.
Schauble, L., Glaser, R., Raghavan, K., & Reiner, M. (1991). Causal models and experimentation strategies in scientific reasoning. The *Journal of the Learning Sciences*, 1(2), 201-238.
Seligman, M. E. P. (1975). *Helplessness: On depression, development, and death*. San Francisco: Freeman.
VanLehn, K. (1988). Toward a theory of impasse-driven learning. In H. Mandl & A. Lesgold (Eds.), *Learning issues for intelligent tutoring systems* (pp. 19-41). New York: Springer-Verlag.
Williams, M. D., Hollan, J. D., & Stevens, A. L. (1981). An overview of STEAMER: an advanced computer-assisted instruction system for propulsion engineering. *Behavior Research Methods and Instrumentation*. 13(2), 85-90.
Ziegler, R. (1992). Worldwide technology firm reaps returns on interactive multimedia training. In *Interactive Multimedia: return on investment analysis for learning and communication* (pp. 13-15). San Francisco, CA: Macromedia.

ACKNOWLEDGMENTS

This research was supported in part by the Defense Advanced Research Projects Agency, monitored by the Air Force Office of Scientific Research under contract F49620-88-C-0058 and the Office of Naval Research under contract N00014-90-J-4117, by the Office of Naval Research under contract N00014-J-1987, and by the Air Force Office of Scientific Research under contract AFOSR-89-0493. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization. The Institute receives additional support from Ameritech and North West Water, Institute Partners.

In addition to the authors, the following people were involved in the development of HeRMiT: Larry Langelier, Donna Fritzsche, Wayne Schneider, Michael Korcuska, Tom Hinrichs, Joshua Tsui, Carolyn Caballero, Leena Nanda, Cheryl Jindra, Matt Greising, Alan Nowakowski, and Don Jastrebsky. Special thanks to the HeRMiT storytellers: Bill Braemer and Tim Coan.

Development and Evaluation of a Series of Hypermedia Educational Systems for the Earth Sciences

Carmelo F. Ferrigno & Denise A. Wiltshire
U.S. Geological Survey
12201 Sunrise Valley Drive, Mail Stop 801
Reston, Virginia 22091
E-Mail: ferrigno@sun4bta.er.usgs.gov

Abstract: Hypermedia technology offers new opportunities for developing earth science teaching materials. During the past three years, the U.S. Geological Survey has conducted a series of research and development projects to assess the effectiveness of hypermedia technology in providing innovative teaching products for middle school students. The goal of the research is to develop interactive computer software that is based primarily on the visualization of earth processes to facilitate learning. As a result of the research, the U.S. Geological Survey has developed two hypermedia products that serve as prototypes of electronic teaching tools. Interdisciplinary teams were formed to design, implement, and test the educational hypermedia software products. The participation of teachers and students in the design and testing phases was critical to the success of the project. Standardized testing procedures were developed during the course of the project to ensure software quality. The combination of manual and automated testing procedures were essential to the success of product development. Preliminary results of the educational hypermedia study indicate that more stringent quantitative and qualitative evaluations of the products are needed. Producing multimedia educational tools for minimally-configured computers continues to pose problems for incorporating high-resolution animation, imagery, video sequences, and sound. The next phase of research conducted by the U.S. Geological Survey will focus on migrating from presentation-oriented hypermedia systems toward interactive applications that empower students with the capability to create their own multimedia notebooks.

In 1991, the U.S. Geological Survey began a series of research and development projects to improve outreach to the pre-collegiate educational community. The goal of the outreach program is to develop a diverse approach to meeting the earth science information needs of the education community. Three program goals are currently defined:

- Materials development (traditional printed teaching packets)
- Experimental computer software applications
- Student-teacher activities, such as workshops and geologic field trips.

The U.S. Geological Survey began the initial computer software project in 1991 to develop an earth science computer system based on hypermedia technology. The objective of the research and development project was to design and implement a hypermedia educational system aimed at middle school students. The project resulted in the production of a hypermedia system known as *GeoMedia* (Wiltshire & Powell, 1993, p. 96) that presented a mix of information on the water cycle, earthquakes, and understanding maps. Each of the three modules contains animations, illustrations, text, a glossary of terms, and a reading list. The application was designed for Apple®[1] Macintosh® computers and developed using MacroMind® Director™ authoring software, which is produced by Macromedia, Inc.

[1] Any use of trade, product, or firm names is for descriptive purposes only and does not imply endorsement by the U.S. Geological Survey.

Teacher Evaluations

GeoMedia was distributed on digital compact disc to earth science teachers in the United States. To date, approximately 1,700 discs have been distributed to educators willing to participate in the project by supplying evaluation comments. Each disc was accompanied by a reader response form that queried teachers on the following evaluation criteria:

- Was the material appropriate for grades 4–8?
- Did *GeoMedia* increase an interest in the earth sciences among students?
- How well did the application perform on your computer system?
- Was the system easy to use and was the supplied documentation helpful?

The project team created a database containing the responses prepared by recipients of the software. Preliminary analysis of the evaluation data show that five basic conclusions may be drawn:

- The material is appropriate for grades 4–8.
- The hypermedia presentation approach is appropriate for a wider range of grade levels than traditional text books.
- Student interest in the earth sciences was increased using *GeoMedia*.
- The performance was slow on Macintosh systems typically found in middle schools (Macintosh LC series).
- In spite of the limitations of system performance on minimally–configured Macintosh systems, students responded favorably to the hypermedia format.

The informal evaluation resulted in a low response by teachers (120 out of 1700 responded). The targeted response by the project team is 30 percent. Although only 7 percent of the teachers have responded to the evaluation form, some *GeoMedia* recipients have written to explain that the hypermedia system will be presented at teacher workshops conducted during the coming year. Another contributing factor to the low response may be that a self–addressed mailer was not included in the shipment with the digital compact disc and evaluation form. The project team plans to prepare follow–up correspondence to teachers coupled with the distribution of volume 2 in the series (*GeoMedia2*). In addition, the project team plans to confer with marketing research experts on how to improve the teacher evaluation questionnaire for the second volume of *GeoMedia*.

Scope of GeoMedia2

Favorable responses to *GeoMedia* from U.S. Geological Survey employees and the education community led to the development of a second hypermedia system on global environmental change. Again, the targeted academic level is middle school. The basic design metaphor of *GeoMedia2* is patterned after the original system (Evans, 1993, p. 69). *GeoMedia2* includes three modules on the carbon cycle, greenhouse effect, and monitoring environmental changes over time. The same graphical user interface is used with a few modifications based on teacher evaluations of *GeoMedia* (Figure 1). The modules each contain four sections: Animation, Elements, Glossary, and Further Reading. The Carbon Cycle module illustrates the movement of carbon through the environment and the effect of human interactions on the cycle. The Greenhouse Effect module explains a natural environmental process that traps heat in the lower part of the Earth's atmosphere to keep our planet warm enough to sustain life. In the Time & Change module, students learn about the geologic history of the Earth and the evolution of living organisms. *GeoMedia2* focuses on the many changes that have occurred throughout the 4.5 billion history of the Earth in the physical, chemical, geological, and biological characteristics of the planet.

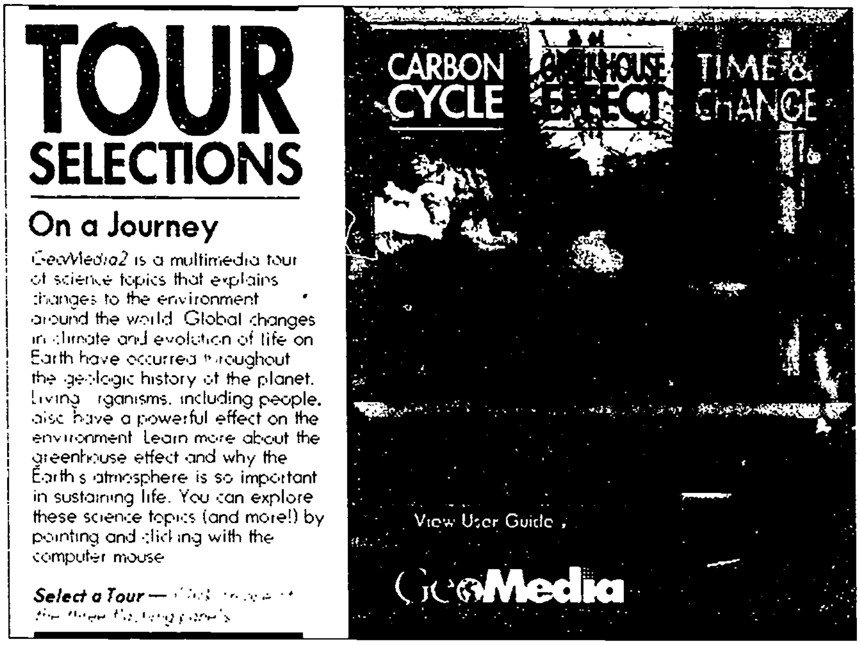


Figure 1. Computer screen from *GeoMedia2* showing table of contents.

Development Issues

Once again an interdisciplinary project team was assembled with a mix of skills in writing, animation production, graphic design, and computer systems analysis to develop *GeoMedia2* (Wiltshire & Ferrigno, 1993, p. 550). Scientists with expertise in global environmental change served as advisors to the project team. Teachers and students played a more active role in the conceptualization phase of the production of *GeoMedia2*.

A workshop was held at the National Science Teachers Association annual convention midway through the project to incorporate suggestions by teachers relating to subject coverage and system functionality. As a result of the workshop, the global change glossary was expanded to include pronunciations of terms. The third module contained in *GeoMedia2* was significantly modified to include more information on geologic time scales and the relationship between environmental changes and species evolution.

Based on discussions with teachers regarding curriculum and subject coverage in *GeoMedia2*, the recommended academic level for this second volume was changed to grades 6–9. Unlike *GeoMedia*, which addresses well established earth science topics, the material presented in *GeoMedia2* is inherently more complex. The information on topics, such as global warming and depletion of the ozone layer, is based on theories currently undergoing testing and analysis. The scientific debate surrounding issues related to global environmental changes contributed to the difficulty in translating theories to accurate material appropriate for the age group. The complexity of the topics also contributed to the longer length of the animations compared to the first volume of *GeoMedia*.

The underlying file structure of *GeoMedia2* was modified to increase performance of the application. The animation and sound files were divided into many small files that load into memory at a more rapid rate. Hence, performance on minimally–configured computers is improved. A file–naming convention was implemented to improve software development and to facilitate testing procedures. The project team also developed typographic standards to improve consistency in text-based navigation instructions.

Additional printing options from within the *GeoMedia2* user interface were added in response to teacher evaluations requesting functionality improvements. The capability to print the *GeoMedia2 User Guide* and student reading lists from within the application was added, thus eliminating the need for he user to initiate printing from the desktop of the digital compact disc (Figure 1). Many of the evaluation responses indicated that teachers did not notice the user guide on the desktop of the digital compact disc. Some teachers confused the configuration information printed on the insert to the compact disc jewel case with the user guide. The option to print the reading lists from within *GeoMedia2* was added to the Further Reading section of each module (Figure 2).

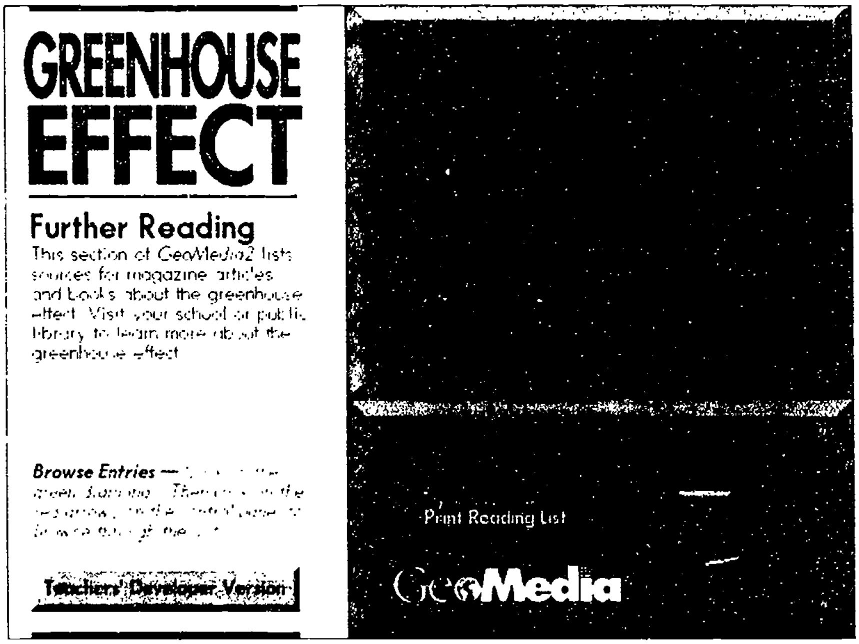


Figure 2. Computer screen from *GeoMedia2* showing print option.

Another user interface design issue related to problems with exiting the *GeoMedia* application. The original purposes of the *GeoMedia* application were for use in the U.S. Geological Survey Visitors Center in a kiosk and for presentations at outreach activities, such as Earth Day celebrations. The design team intentionally omitted an icon to represent the quit function to guard against students escaping from *GeoMedia* to the Macintosh desktop. Two keyboard sequences are available for exiting the application and are described in the user guide. Although the method for exiting the program was not changed for *GeoMedia2*, specific instructions on exiting the application were added to the printed material contained on the insert to the compact disc jewel case.

To further improve functionality, the project team investigated the QuickTime™ system extension for use in *GeoMedia2* as a compression algorithm for video sequences and animations. QuickTime can also be used to compress large images, which reduces the amount of memory required to store and display images (Don, 1992, p. 52). Three factors that affected use of QuickTime in *GeoMedia2* include: (1) the system extension is proprietary and must be obtained separately from the Macintosh operating system, (2) the system extension must be installed in the user's configuration file (Macintosh System Folder), and (3) the image resolution and frame rate for video sequences and animations are low. Use of QuickTime in *GeoMedia2* is restricted to two video sequences that show global climate models generated by supercomputers and digitized from a video tape. If the user does not have QuickTime installed, a still image appears instead of a video sequence. The optional display of video sequences offers flexibility when developing hypermedia applications for both low– and high–performance microcomputers. As noted above, replacing video sequences with still images or animated sequences ensures reliable performance on minimally–configured computers.

Testing Procedures

Extensive colleague reviews and functionality tests were required to ensure the scientific accuracy of the information and the reliability of the application. An advisory group of scientists from the U.S. Geological

Survey conducted colleague reviews of both *GeoMedia* applications to verify the accuracy of the information.

Teachers played a more prevalent role in the beta testing process of *GeoMedia2* than for the first volume. In addition, a revised testing procedure was developed for *GeoMedia2* to improve the ability to duplicate or trace errors, such as incorrect links. The original testing procedure was based on an outline of links found in the main screen for each module. This procedure was completely manual and marginally satisfactory for conducting reliable tests. For example, if an error such as an incorrect link was discovered, it was difficult to retrace the path taken through the application because no manual or automated procedure recorded this path.

Because of time constraints and limitations of the authoring system used for *GeoMedia2*, no automated testing procedures were developed. However, an improved testing document was developed to include a snapshot of each screen of the application. All navigation icons and links were represented in the testing document along with tables to represent paths through the information. The tables served as checklists for the technical project staff and teachers conducting the beta tests. The outline used for beta testing volume 1 of *GeoMedia* did not provide a formal mechanism for indicating text changes. Snapshots of each screen provided in the *GeoMedia2* testing document allowed reviewers to easily note text changes. The expanded testing document provided an excellent method for correcting errors in the text and for consistently documenting software problems. The testing document continued to provide a marginally acceptable method for testing non-sequential navigation patterns. Hence, the testing procedure remained linear.

To thoroughly test the application in a non-linear method, an automated procedure is required. Using an automated approach for capturing navigation data provides an accurate report of the reviewer's path through the information. The navigation data or audit trail can facilitate the testing and correction process (Beasley, 1992, p. 466).

Future Plans

The U.S. Geological Survey plans to continue assessing the effectiveness of hypermedia technology in teaching earth science concepts. Plans are in place for developing a third educational system that provides students with an interactive application that extends beyond a presentation format by empowering the student to create interpretations of the information. While *GeoMedia* has received favorable review comments, a need exists to develop a hypermedia system that enables students to navigate through a non-sequential arrangement of information and further participate in the creative process by authoring their own multimedia reports (Kinnaman, 1990, p. 46).

The proposed application will explain earth science processes within a geographically–referenced hypermedia system. Students will explore earth science topics by selecting geographic regions that are notable for extreme (unique) hydrologic or geologic conditions. The approach to the information will be by way of maps and satellite images instead of topics from a table of contents, as in *GeoMedia*.

Summary

As a result of a three–year study, the U.S. Geological Survey has developed two hypermedia products that serve as prototypes of electronic teaching tools. Interdisciplinary teams are needed for design and conceptualization of the products in addition to testing. The participation of teachers and students in the design and testing phases is also critical to the success of the project. Standardized testing procedures are required to ensure software quality. The combination of both manual and automated testing procedures is essential to the success of product development. Preliminary results of the educational hypermedia study indicate that more stringent quantitative and qualitative evaluations of the products are needed. Producing multimedia educational tools for minimally–configured computers continues to pose problems for incorporating high–resolution animation, imagery, video sequences, and sound. The U.S. Geological Survey plans to migrate from presentation–oriented hypermedia systems toward interactive applications that empower students with the capability to create their own multimedia notebooks.

References

Beasley, R. E. (1992). A methodology for collecting navigation data using IBM's LinkWay–A hypermedia authoring language. *Journal of Educational Multimedia and Hypermedia*, 1 (4), 465-470.

Don, Abbe. (1992). QuickTime makes movies on the small screen. *Publish*, 7 (5), 50-57.

Evans, Poppy. (1993). Interactive geology. *Step-By-Step Graphics*, 9 (5), 68-77.

Kinnaman, D. E. (1990). The next decade—What the future holds. *Technology & Learning*, 11 (1) 43-49.

Wiltshire, D. A., & Ferrigno, C. F. (1993). GeoMedia: A hypermedia system on earth science topics for middle school children. In *Educational Multimedia and Hypermedia Annual. 1993, Proceedings of ED-Media 93—World Conference and Educational Multimedia and Hypermedia*, Orlando, Florida, June 23-26, 1993. Charlottesville, Virginia, Association for the Advancement of Computing in Education, 549-555.

Wiltshire, D. A., & Powell, M. E. (1993). GeoMedia—Teaching earth science through new technology. In *U.S. Geological Survey Yearbook. Fiscal Year 1992*. Reston, Virginia, U.S. Geological Survey, 96-98.

Acknowlegements

The authors give special thanks to the teachers and professors who assisted in the design and development of the GeoMedia educational software. We are especisally grateful to the following educators who were an integeral part of the beta testing team: James Edson, University of Arkansas at Monticello; John Kinnear, Turlock Junior High School, Turlock, California; Russell Marolf, Odyssey School, Rochester, New York; David Parisian, Oswego High School, Oswego, New York; and Margo C. Sterling, Dogwood Elementary School, Reston, Virginia.

Which Way Will The Wind Blow?
Networked Computer Tools For Studying The Weather

Barry J. Fishman and Laura M. D'Amico
School of Education and Social Policy
2115 N. Campus Drive, Northwestern University, Evanston, IL 60208
E-Mail: bfishman@ils.nwu.edu, damico@covis.nwu.edu

Abstract: A suite of networked computer tools within a pedagogical framework was designed to enhance earth science education at the high school level. These tools give students access to live satellite images, weather maps, and other scientific data dealing with the weather, and make it easy for students to make their own weather forecasts by creating high-quality weather maps. These tools are part of the Learning Through Collaborative Visualization (CoVis) project, an advanced technology research and development effort to explore the effects of giving high school students access to many of the same visualization and electronic communication tools used by scientists.

Science education in most contemporary high schools is far removed from the actual practice of science. Students are not given the opportunity to understand what comprises the real work of scientists and as a result are not attracted to science professions. School presents science as a series of known quantities. Student projects, when they exist, recreate answers to historical problems. By contrast, scientists spend most of their time researching questions for which there currently are no answers. In addition, scientists spend a significant amount of time communicating and collaborating with their peers. In this paper, we will describe an approach to studying meteorology that gives high school students access to the same tools and data used by professional and academic meteorologists to forecast and study the weather. This enterprise takes a holistic view of the classroom and its relation to the world outside. To this end, we link computer-based network tools for accessing and manipulating weather data with significant changes in pedagogy for teachers and students.

The Learning Through Collaborative Visualization (CoVis) project at Northwestern University is designed to reconceptualize and re-configure high school science education. A primary goal of CoVis is to enable project-based approaches to science by using computer networks to put students in direct contact with practicing scientists and their scientific tools (Pea, 1993; Pea and Gomez, 1992). A major criticism of current science education (and education in general) is that it isolates students from the world of practice, teaching them decontextualized skills that they are unable to apply in appropriate settings (e.g. Brown, Collins, & Duguid, 1989; Lave & Wenger, 1991). Traditionally, K-12 science education has consisted of teaching well-established facts. It bears little or no resemblance to the question-centered, collaborative practice of real scientists. CoVis is an attempt to transform science learning to better resemble the authentic practice of science. The CoVis tools for studying the weather are crafted to make this possible.

The CoVis Learning Environment

Weather As Subject Matter

Most people have some intuitions about weather. They talk about it in casual conversations with strangers. They stand out on their porch in their bathrobe for five minutes while deciding whether or not to wear a sweater or carry an umbrella. Although they listen to the television weather man, they know that even with all his maps and charts, he is often wrong.

Furthermore, there has been an increasing interest in weather at all levels of society, perhaps prompted by the dramatic weather events of the past two years. On the Internet, resources related to weather have long been popular. Two heavily used resources for on-line weather data are the "Weather Underground" at the University of

Michigan and the "Weather Machine" at the University of Illinois at Urbana-Champaign (UIUC). The Weather Underground receives over 100,000 connections in an average week (Weather Underground, 1993), and the Weather Machine receives over 30,000 requests for information each day (Ramamurthy & Kemp, 1993).

By the time high school students begin to study weather, they have developed a naive sense of how to predict the weather. For the most part, they rely upon a combination of what they see when they look up at the sky and what the media tells them to expect that day. But what if they had access to the same weather data and images that the forecasters have? Imagine them looking at current on-line weather maps which can zoom in to show local weather patterns and zoom out to show all of North America. What if they had the ability to create their own satellite animations? Then imagine them using all this information and more to make their own predictions of what the weather will be like later today, tomorrow or even next week, creating their own maps and then comparing their predictions to what really happened. All of this is possible using a set of tools developed as part of the CoVis project. These tools are known as the Weather Visualizer and Weather Graphics Tool.

The Technological Setting for the CoVis Weather Tools

The CoVis project is currently working with two high schools in the Chicago area, providing them with technology and curriculum support (for a more detailed description of hardware in the CoVis classroom, see Note 1). Each CoVis computer is a Macintosh with a direct (TCP/IP) Internet connection, with standard Internet client software for electronic mail, Usenet news, FTP, and Gopher. Specialized visualization tools created by the CoVis research team take advantage of Internet connectivity to provide current weather data. The tools used by scientists, although powerful, are somewhat cumbersome for students to use. To facilitate student work, the CoVis team designed front-end software that assists students in making connections between the visualizations and the phenomena they represent (Gordin, Polman, & Pea in press).

Three specialized communication tools are provided to allow students located in different classrooms to work collaboratively on projects. The most significant tool in this respect is Cruiser, a desktop video telephone system developed at Bellcore (Fish, Kraut, Root, & Rice, 1993). Using this tool, students can hold joint project meetings across school sites. A commercial screen sharing tool from Farallon Computing called Timbuktu is used to allow students to work collaboratively on the same piece of computer software. The combination of desktop video teleconferencing and screen sharing software allows students to do synchronous collaboration and consultation with students at both schools, atmospheric scientists at UIUC, or museum and library staff at the Exploratorium in San Francisco, CA. Finally, a specialized "Collaboratory Notebook" was developed to allow for the recording of projects and the storage and retrieval of jointly created artifacts, such as weather maps (Edelson & O'Neill, 1993). Considered together, these communication tools expand the CoVis learning environment to encompass information resources, activities and colleagues far beyond the four physical walls of the classroom.

The Pedagogical Setting for the CoVis Weather Tools

Student use of the technological tools described in this article both enables and is enabled by the pedagogical context of the CoVis classroom. Science education as it currently exists in the United States removes the subject matter from real contexts. Even in-class science labs, which are intended to give students "hands on" experience, are frequently no more than cookbook procedures. The result is that students "learn little... about the conduct of science; [reinforcing] the idea that science is dull, procedural, and thoughtless" (Tinker, 1991). If we consider the traditional approach to science education as learning-before-doing or learning-outside-doing, then the best response is to design educational systems and supporting technologies that allow learning-in-doing (Pea, 1993).

Learning-in-doing in the classroom is best exemplified by the project method of doing science (Ruopp, Gal, Drayton, & Pfister, 1993). Implicit in the adoption of the project method is a view that learners need to discover and develop facts and skills for themselves, as opposed to having material "delivered" via lecture or other kinds of presentation. Another important assumption is that learning must be located in communities of practice. This is also reflected in writing and thinking on cognitive apprenticeship (Collins, Brown, & Newman, 1989; Brown, Collins, & Duguid, 1989). What materials one chooses for learning is closely connected to notions of apprenticeship learning. For the CoVis project, we provide on-line access to scientific data and tools because we believe that these sources are more authentic than textbook information about science.

Finally, the role of the teacher must shift. Where previously the teacher was seen as a source of information, he or she now must become a source of support and guidance. Collins, et al. (1989) refer to this as a coaching role for the teacher. In considering the pedagogical bases of the CoVis classroom, it is important to remember that technology is not required to achieve this kind of classroom structure. Technology, however, enhances what can be accomplished in such a classroom and provides a lever to assist the teachers in changing their own approach to the classroom. Certainly, without the pedagogical changes made to the CoVis classroom, the uses for the technologies would not be as interesting or productive.

A Scenario for Studying Weather

How do high school students currently think about weather? Most people have a naive understanding of the weather, built up from their own life experience. High school students gain additional insight from media sources like the Weather Channel, the daily paper, or the nightly television news broadcast. In the CoVis project, we believe that these naive conceptions are an ideal place to begin the learning process. The tools we provide are designed to support and augment an inquiry curriculum where students learn to ask and answer their own questions about atmospheric science. Through such investigations the students are able to expand their knowledge and understanding of atmospheric phenomena beyond their localized daily experience of weather conditions. They incorporate explorations of larger weather patterns, the effect of geographic formations on weather, activity and conditions in the upper atmosphere, along with a more diverse number of atmospheric measurements, to create sophisticated mental models of how weather works. The following scenario describes each of the Weather Visualizer components as they might typically be used in the classroom.

Building From Naive Knowledge

The best starting point for an exploration of weather phenomena is the sky that a student sees outside his or her window. In one of the CoVis classrooms, the teacher asks his students to spend five minutes two times a day looking at the sky, and writing down what they observe in their journals. The students are instructed to record cloud cover and current weather conditions. Additionally, they are asked to make a prediction for what the weather will be like the next time they make an observation. Over a period of about two weeks, the students begin to form a reasonably accurate (though naive) predictive ability.

Observation-based forecasting of this kind is an ancient art dating back at least six-thousand years. During this time some rather accurate weather folklore or "weather wisdom" has developed (Lee, 1976). The CoVis teacher explains some of this folklore to his students, asking them to make use of it in their own predictions where appropriate. Students explore and compare their own predictions to those indicated by the weather lore. By viewing these comparisons in light of how the weather actually unfolds, the students are able to critique both their own and the weather lore's predictive merits.

By its very nature, observation-based forecasting is useful only for short-term local predictions of small-scale phenomena. To understand weather patterns on a larger, more integrative scale, students must be able to collect and analyze weather data from outside their own personal experience. When the teacher feels that the students have formed their own beliefs about local weather prediction, he begins to expand their knowledge base of weather experience by introducing them to our Weather Visualizer.

The Weather Visualizer is a suite of tools that simplify access to information provided by the University of Illinois via its "Weather Machine." That network resource provides a broad array of satellite images, weather maps, and specialized scientific visualizations of weather data to Internet users. Our Weather Visualizer makes access to those tools easier; it simplifies the Weather Machine's interface; and it extends functionality in several significant ways designed to meet specific pedagogical needs.

The Weather Visualizer is a client/server technology that is increasingly common in university and industrial models of "distributed" computing. The tool itself resides on students' Macintosh computers. When various commands are issued, the Macintosh sends an information request to a Sun workstation at Northwestern. If the machine at Northwestern contains the necessary information, it is sent back to the Macintosh where it is displayed. If, however, more or updated data is needed, Northwestern's computer sends an information request to the Weather Machine at UIUC. All of this network access takes place in the background, without any student intervention. The student is thus better able to focus on the task of thinking about the weather, instead of thinking about network access tools.

Changing Perspective With Satellite Images

The images most students choose to look at first are satellite images (see Figure 1). Using a push-button palette interface, the Weather Visualizer provides both visual and infrared satellite images of the United States as taken by the GOES-7 weather satellite in near-real time. These images are an ideal complement to student's own observations. The same clouds that the students observed from the ground are now seen from Earth orbit — an engaging shift in perspective.

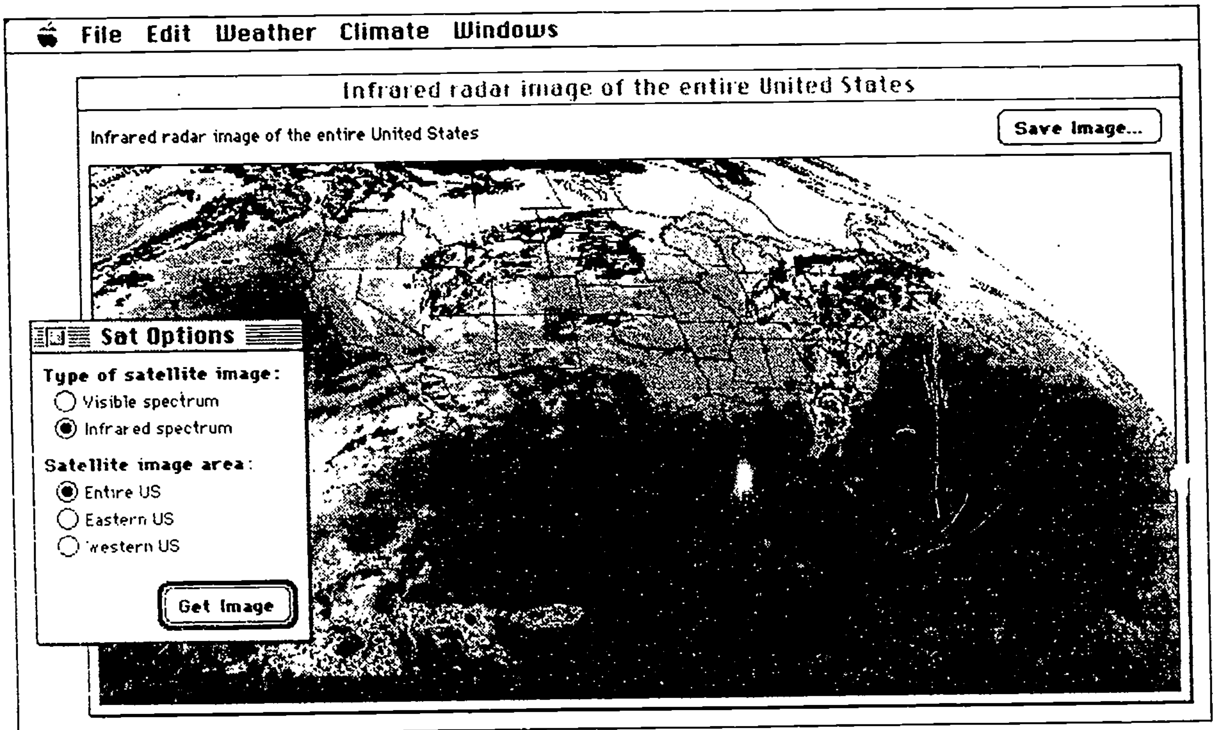


Figure 1. A satellite image from the Weather Visualizer with the palette used to generate current images.

The satellite images extend a student's view dramatically. Not only can they see the cloud cover above their current location, they can also see cloud cover over the rest of the nation. The infrared images allow students to gauge how high the cloud tops are, a good indicator of potential precipitation. If the student animates a series of images, those clouds can be set in motion.

Using the Weather Visualizer, students can make forecasts for areas beyond their own location. Large-scale weather patterns, such as mid-latitude cyclones, become apparent. Students begin tracking and making predictions about the formation and life cycle of these large-scale weather patterns. Though not apparent from local observations, they nonetheless affect local weather phenomena. Students compare the large-scale weather pattern predictions to their local weather data collection and prediction efforts and begin forming correlations between their local weather experience and the large-scale weather patterns.

Student-Customized Weather Maps

The information in a satellite image is a direct representation of atmospheric conditions. This directness, however, means that the students must make their own interpretations of what is visible in order to make predictions. There is no information present in the satellite image about precise temperatures, dew points, surface conditions, or wind speed and direction (although the last two could, in a rough fashion, be calculated from an animation). At this point the teacher continues to build on the student's naive understanding by introducing another representation that includes this additional information. WxMaps are specialized weather maps produced by software written at the University of Illinois. These maps are generated by a UNIX program using data from the National Weather Service called Domestic Data Plus (DD+). The UIUC software, although very powerful, has a difficult to use command-line interface. The Weather Visualizer puts a graphical front end on this UNIX software, allowing students to construct their own WxMaps by checking off various options (see Figure 2).

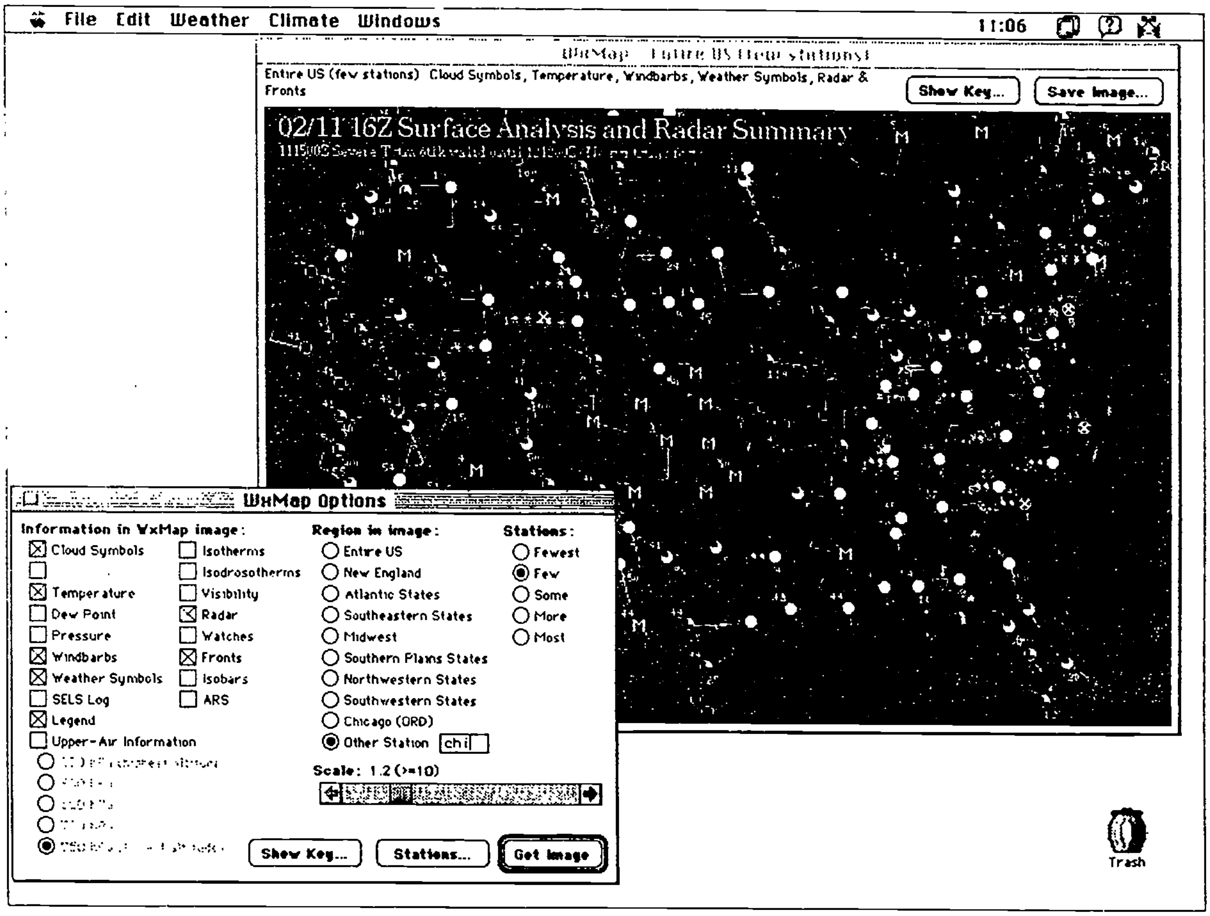


Figure 2. A customized weather map with the palette used to generate current maps.

The WxMap images display a variety of conditions at each reporting National Weather Service station in North America, and interpolated data generated from all the stations (for a complete listing of variables provided by the WxMap software, see Note 2). Students can focus their observations by region or city, select the density of reporting stations, and zoom in or out on their area of focus. By default the WxMap tool displays current conditions, but data is available from the past forty-eight hour period.

The station information provided by WxMaps allows students to study weather data across the continent that are similar to what they are collecting at their own location. They can explore the relationship between large-scale weather patterns and local weather conditions at various points in North America. As they explore weather conditions in other locations, students consider the effects of local geographic formations on weather phenomena (e.g., lake effect snow in Chicago) on their predictions.

Moreover, the teacher asks the students to expand their thinking about weather from surface effects to patterns in the upper air. WxMaps are capable of showing upper air conditions, such as jet stream wind patterns, using data gathered by weather balloons released twice a day at each station, The teacher asks the students to compare these upper air maps to the surface maps so that they may begin to think of the atmosphere as a three-dimensional space. Through exploring this three-dimensional space, the students learn that patterns of atmospheric phenomena occur not only in horizontal directions, such as the general west-to-east movement of the jet-stream over the U.S., but also in vertical patterns such as rising and sinking masses of cold and hot air.

Making Personalized Maps with the Weather Graphics Tool

With the finer-grained data provided by the WxMap tool, students begin to make more detailed predictions of the weather. The CoVis teachers, seeking ways to support this prediction activity, requested a tool that would allow students to make maps representing their predictions. In response, the CoVis team developed the Weather Graphics Tool. This tool is actually a plug-in extension to Aldus SuperPaint 3.5, a widely used graphics tool for the Macintosh. Our plug-in provides SuperPaint with a palette containing "stamps" for all the weather symbols represented in WxMap. We also devised a number of blank maps for students to use, one map for each region plus national maps, half with and half without weather station names. These maps are stored as Macintosh stationery pads.

To use the Weather Graphics Tool, a student selects a base map and uses it to launch SuperPaint. Then, using the plug-in palette, students select various weather symbols and place them on the map. There is a special palette for constructing the standard station model (see Figure 3), which is then placed by clicking on the map like any of the other stamps. After constructing the weather map, the student can store it for later comparison to a map generated by WxMap. The ability to create maps using the Weather Graphics Tool is motivating to students because the maps they create are as professional looking as those generated by the WxMap software.

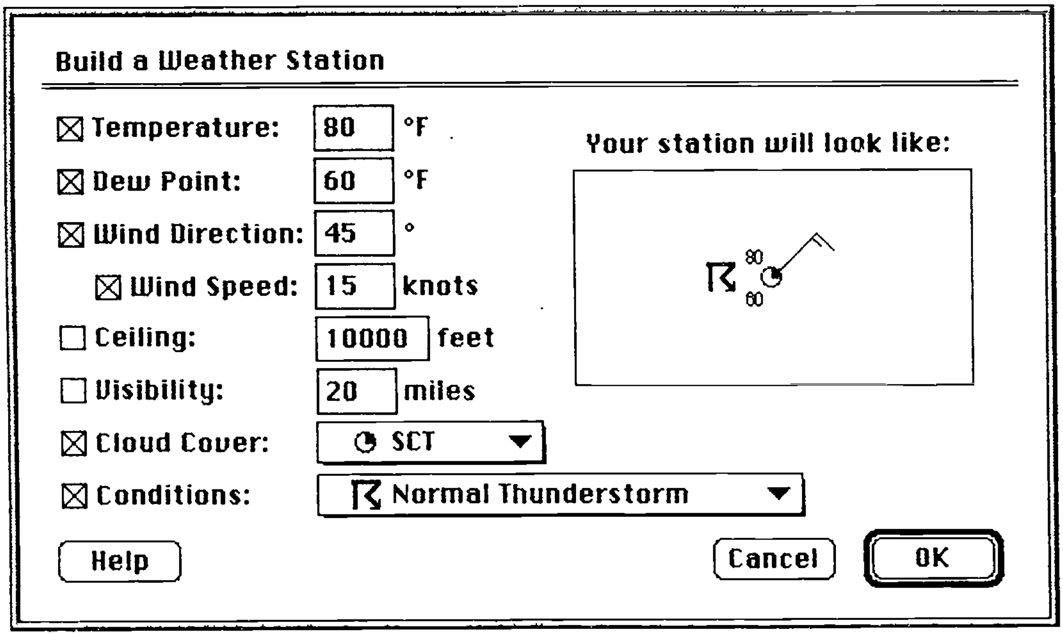


Figure 3. The palette used to generate a standard station model in the Weather Graphics Tool.

In order to make such detailed predictions, the students must consider the relation of various atmospheric conditions and processes to future weather at any given time and location. To do so, they must begin to integrate the knowledge they have gained thus far, and like professional weather forecasters, make decisions about which weather indicators have more weight in any given weather scenario.

Teachers like the Weather Graphics Tool for the role it plays in prediction activities, but they also envision other uses for it. For example, it is very easy to create weather "scenarios" using the Weather Graphics tool and ask students to explain what they see. The tool allows teachers to easily create impossible scenarios, which require students to explain why a certain combination of phenomena could not exist in reality. Also, maps generated using the Weather Graphics Tool are easily added to students' evaluation portfolios.

Advanced Interpretation — Six-Panel Images

As students learn how to make more accurate weather predictions, they are also learning more about the representational language of atmospheric science. This learning is driven by the higher-level goal of improving their forecasts, and thus happens in the background of the learning activities. For students who have mastered the interpretation of WxMaps, there is one more type of image available from UIUC. This image, commonly referred to as a "Six Panel" image, displays six atmospheric variables mapped as color across a map of the United States. These six variables are temperature, dew point, pressure, wind direction, wind velocity, and moisture convergence (see Figure 4). The six panels represent a great deal of information presented in an efficient manner. As a result, they are difficult for novices to interpret. There are, however, direct benefits to investing the time to learn how to interpret the Six Panel image. Moisture convergence is a particularly useful variable, because it indicates where there are gathering "pools of moisture" in the atmosphere that could contribute to future precipitation.

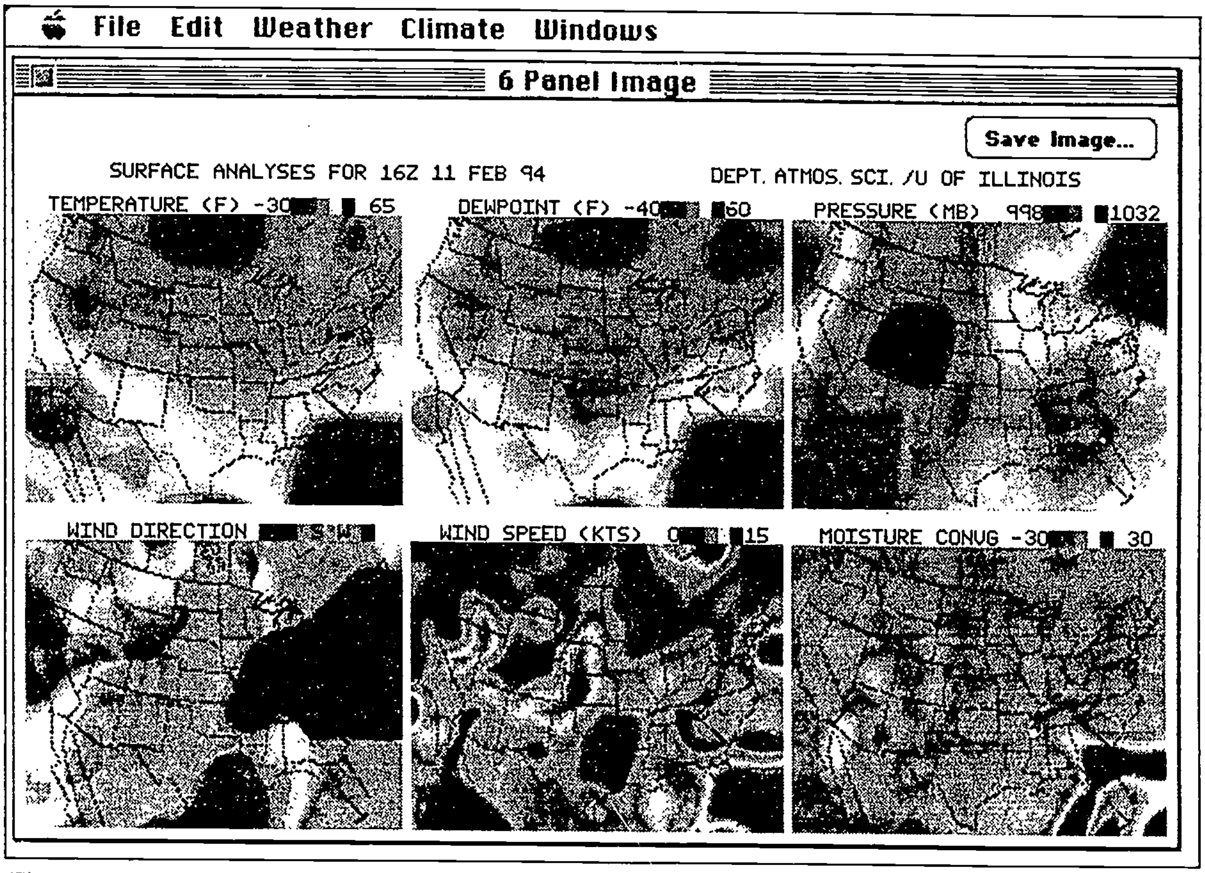


Figure 4. A six-panel image showing temperature, dewpoint, pressure, wind direction and speed, and moisture.

Conclusion

When students have the ability to work and think with the same tools that scientists use, remarkable things can happen in the classroom. Topics that were previously dry become animated and exciting for students. Instead of memorizing the station model and its interpretation as facts to be repeated on a test, students put the station model to use in illustrating forecast maps that they construct using the Weather Graphics Tool. They can look at live satellite photos and animations for a unique perspective on the sky. WxMaps and Six Panel images make it possible to understand a broad array of reports from weather stations across the country. When tools such as these are used to build upon what they already know, students have a starting point for exploration that is internally motivating.

The combination of the CoVis weather tools and the pedagogical approach of the CoVis classroom links the learning of science more closely with the practice of science. Students in the CoVis classrooms learn more than science, they learn what it is like to do science. This is an important step on the path to making science education both more relevant and enjoyable.

Notes

(1) In the 1993-94 school year the CoVis project involves 282 students in twelve classes at two Chicago-area high schools. There are six teachers: four Earth Science, one Environmental Science, and one Science, Technology, & Society. Each classroom (one per school) is equipped with five Macintosh Quadra 700 computers with 8mb of RAM and 16 inch color monitors, one laser writer, and video projection equipment. Each Quadra has a Cruiser station next to it, composed of one 13 inch NTSC television monitor, microphone, speaker, and wide-angle NTSC television lens. There is one additional Quadra at each high school for students to access outside of their class hours. Internet and video network connections for both high schools are provided by primary-rate ISDN connections to Northwestern's network backbone and the public-switched telephone network.

(2) The WxMap software reports the following variables: clouds, station names, temperature, dew point, pressure, wind velocity, wind direction, current weather conditions, visibility, radar, watches, fronts, isobars, isotherms, and isodrosotherms. Upper air analyses are available at 850, 700, 500, 300, and 200 hPa. Additionally, the WxMap software can display the Automated Radar Summary and the daily Severe Local Storms (SELS) log, both products of the National Weather Service.

References

Brown, J.S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher, 18*(1), 32-42.

Collins, A., Brown, J.S., & Newman, S.E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L.B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Erlbaum.

Edelson, D.C., & O'Neill D.K. (1993). *The CoVis Collaboratory Notebook: Supporting collaborative scientific inquiry.* Paper presented at NECC '94, Boston, MA.

Fish, R.S., Kraut, R.E., Root, R.W., & Rice, R.E. (1993, January). Video as a technology for informal communication. *Communications of the ACM*, p. 48-61.

Gordin, D., Polman, J., & Pea, R.D. (in press). The Climate Visualizer: Sense-making through scientific visualization. *Journal of Science Education and Technology.*

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* Cambridge: Cambridge University Press.

Lee, A. (1976). *Weather Wisdom: Facts and Folklore of Weather Forecasting.* New York: Congdon and Weed.

Pea, R.D. (1993, May). Distributed multimedia learning environments: The Collaborative Visualization Project. *Communications of the ACM*, p. 60-63.

Pea, R.D., & Gomez, L.M. (1992). Distributed multimedia learning environments: Why and how? *Interactive Learning Environments,* 2(2), 73-109.

Ramamurthy, M.K., & Kemp J.G. (1993, October). Using the Weather Machine: A gopher server at the University of Illinois brings weather information to every desktop. *Storm: The World Weather Magazine*, p. `4-39.

Ruopp, R.R., Gal, S., Drayton, B., & Pfister, M. (1993). *LabNet: Toward a community of practice.* Hillsdale, NJ: Erlbaum.

Tinker, R. (1991). *Thinking about science.* Unpublished manuscript, T.E.R.C., 2067 Massachusetts Avenue, Cambridge, MA, 02140.

Weather Underground. (1993). *Weather as the paradigm for instructional technology.* Department of Atmospheric, Oceanic, and Space Sciences, University of Michigan, Ann Arbor.

Acknowledgements

The authors would like to thank Joey Gray for his programming and design work in the development of the CoVis weather tools, and Eric Baumgartner for his work on the Weather Graphics Tool. Thanks to Roy Pea, Louis Gomez, Danny Edelson, and Steve Hall for their comments on earlier drafts of this paper. We gratefully acknowledge Aldus Software for providing SuperPaint to the CoVis project, Apple Computer, Farallon Computing, Sun Microsystems, and the support of National Science Foundation Grant MDR-9253462 in work related to this paper.

Critiquing the computer-aided design of dental prostheses

F. J. Fitzpatrick[1], J. C. Davenport[1], P. Hammond[1,2] and D. A. Randell[1]

[1]*The School of Dentistry, The University of Birmingham, St. Chad's Queensway, Birmingham B4 6NN, England. E-mail: f.j.fitzpatrick@uk.ac.bham.*
[2]*Advanced Computation Lab, Imperial Cancer Research Fund, Lincoln's Inn Fields, London.*

Abstract: This paper describes RaPiD, an assistant for the design of dental prostheses called removable partial dentures. The user manipulates icons directly to indicate the desired design solution to a given clinical situation. A developing design is represented as a logic database of components in a design; expert rules are applied as integrity constraints governing valid database transactions/design alterations. Contravention of design rules is presented to the user in a critiquing style. The critiquing strategies form the basis for the system's use in undergraduate and graduate dental education.

The education of dental undergraduates includes instruction in the principles of the design of dental prostheses. The design process is a complex and difficult task, and the time available in the dental curriculum to impart the required knowledge is limited. If students were to have open access to a knowledge-based system for the design of prostheses, there could be important educational benefits. Such a system, RaPiD (*R*emovable *P*artial *D*enture design using *a*rtificial *i*ntelligence), is now being developed by specialists in prosthetic dentistry and knowledge-based systems. RaPiD is usable both as a prescription system for dental practitioners and as a teaching system which can be configured for undergraduate and postgraduate use. RaPiD as a prescription system has already been described in the dental and AI literature (Hammond et al, 1993a and 1993b); this paper concentrates on its educational use, in particular on its critiquing strategies which guide a user through a design.

1. Removable partial dentures

A removable partial denture (RPD) is a denture provided for a patient who has some natural teeth remaining. It enables its wearer to chew food effectively, assists speech and helps to stabilise the remaining natural teeth; it may also enhance the patient's appearance. An RPD comprises up to 40 components, the most important of which are as follows. The *saddles* carry the artificial teeth and fit over the area of the gum from which natural teeth are missing. *Rests* are metal extensions which transmit biting forces from saddles to the adjacent natural teeth. *Clasps* are flexible metal clips which grip natural teeth, making a denture secure during function. Finally, a *major connector* is a rigid metal bar or plate which unites the other components into a single prosthesis.

2. The motivation for the development of RaPiD

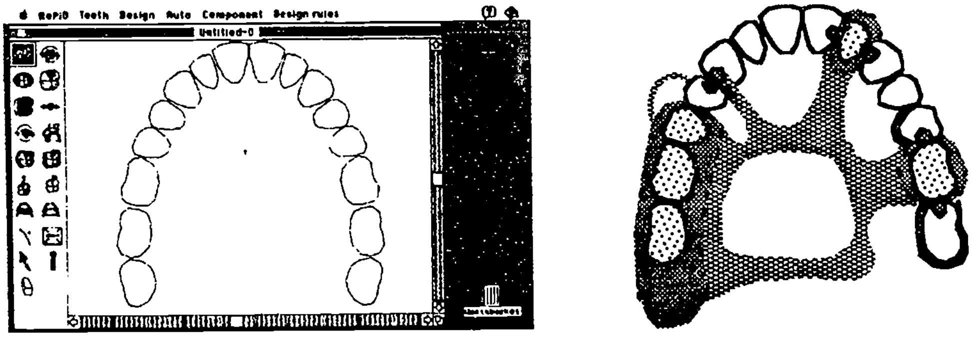
Designing a satisfactory RPD requires clinical training and experience. The dentist should carry out a detailed assessment of the patient and of models of the patient's teeth and then draw a design to be sent to a dental technician as a prescription for the manufacture of the denture. However, general dental practitioners commonly delegate responsibility for design to technicians, whose training is inadequate to assess the clinical factors in a case. The resulting denture may fit poorly, and may even pose a threat to the patient's remaining natural teeth. The reasons for this delegation of responsibility are not entirely clear, although limited design experience may be crucial (Basker et al, 1978 and 1991). One factor which motivated the development of RaPiD was the desire to build an educational tool that could enhance the training of dental students and practitioners. A general dental practitioner in the UK designs, on average, one RPD a month, while in dental schools students may have to produce only five or six RPD designs for patients throughout their undergraduate study (Holt et al, 1993). It is doubtful whether this frequency is sufficient to ensure adequate familiarity with good design principles, whether by students or by graduate dentists. Clearly a knowledge-based system which uses a graphical interface allowing the creation of new designs and alteration of existing ones, which incorporates all the principles governing correct design of RPDs and can detect and critique attempts to violate them, could be a powerful educational tool. The evidence is, then, that a system such as RaPiD will meet a genuine and pressing need.

3. Levels of operation of the system

RaPiD has two modes of operation, manual and automatic. The manual mode is designed for educational use while the automatic version is intended for dentists in practice. The manual version can be run at two levels of difficulty, "student" and "expert". In "student" mode a user who infringes a design rule is informed of the infringement immediately so that the system controls the user's progression through the design sequence towards an optimal solution. In "expert" mode the designer has a free hand: the design is developed by the user without any interference from the system and without any design infringements being signalled. However, on completion of the design the result is appraised by RaPiD, rules that have been broken are listed and alternative solutions displayed. In automatic mode the dentist enters key clinical information, after which the system takes over and completes the design. In all modes, editing of the final design is possible, and a graphical and textual summary of the result can be printed, as a record for the student, for evaluation, or as an instruction to a technician.

4. Designing with RaPiD

Software development has taken place on Macintosh computers, using an implementation of Prolog, MacProlog (Logic Programming Associates, Ltd, London), which provides powerful object-orientated graphics. A prominent feature of RaPiD is its graphical interface which utilises the window-based environment standard in Macintosh applications. Figure 1(a) depicts a design window at the beginning of a design session, with a palette of tools at the left and the design area, containing a dental arch, in the remainder of the window.



(a) Initial screen showing palette of tools (left) and design window with upper arch of teeth.

(b) A completed design for an upper arch.

Figure 1

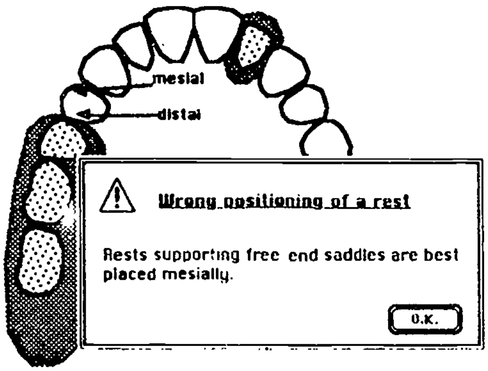
A user begins a design by selecting the "forceps" tool () and then clicking once on teeth icons to be made artificial, whereupon they become shaded. A double-click removes a tooth icon entirely when a tooth is to be missing and not replaced by an artificial tooth. The saddle-placement tool () creates a saddle around the artificial teeth; another tool () places rests on teeth for support of saddles. There are tools also for placing different types of clasps on retaining teeth. The major connector which joins all the components of a denture has to be drawn by the user, employing a mouse. Other tools include one for giving information about individual components, and an editing tool for changing the shape of components. A design can be printed, together with a written summary, for sending to a dental technician for manufacture. Figure 1(b) shows a completed design.

5. The use of critiquing

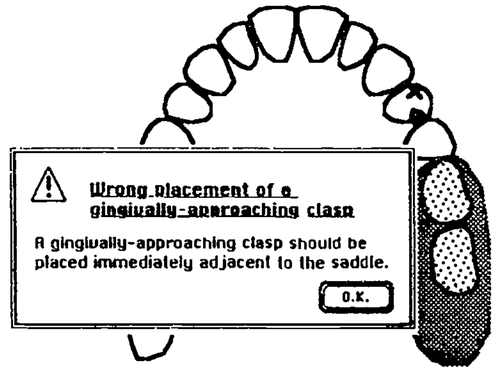
The student of RPD design needs to be made aware of any design errors immediately and to be challenged to modify the design correctly. RaPiD achieves this through a range of critiquing strategies which alert the user to an error, efficiently but as unobtrusively as possible. Three examples will illustrate the use of critiquing.

(i) *Placing rests on supporting teeth.* A saddle which is bounded at both ends by healthy natural teeth is supported by rests placed on the neighbouring teeth. However, with a free-end saddle, one bounded at only one end by natural teeth, this is impossible. Should the one supporting rest be placed on the neighbouring tooth mesially (at the end closer to the front of the mouth) or distally (towards the back)? Prosthetic specialists favour

mesial placement. An attempt to place a rest distally therefore fails and a critiquing message is presented, specifying what is wrong with the proposed alteration and advising correction (Fig. 2(a)). As with other rules, if exceptional clinical circumstances make the rule inappropriate, it can be overridden by means of a function key.
(ii) *Placing a gingivally-approaching clasps.* A gingivally-approaching clasp is a type of clasp which reaches the undercut region of a tooth by passing from the saddle via the gingiva or gum. It is placed by drawing a line connecting the undercut on the gum area of a tooth with a point on the saddle to be clasped. (In Figure 1(b) there is a gingivally-approaching clasp connecting the saddle which carries three artificial teeth with its neighbouring natural tooth.) The clasp which retains a saddle should be placed on the tooth adjoining the saddle. So if a user attempts to draw a clasp starting from any tooth which does not neighbour a saddle, the alteration immediately fails. An error message directs the user to place the clasp on a neighbouring tooth (Figure 2(b)).



(a) Incorrect placement of a rest: the user has attempted to place a rest distally; it should be placed mesially. (Mesial and distal locations not marked in actual design.)



(b) Incorrect placement of a gingivally-approaching clasp: the user has attempted to draw the path of the clasp starting from a tooth (marked with "X") which does not neighbour a saddle.

Figure 2 Critiquing messages

(iii) *Drawing a major connector.* A major connector should be added to a design only if all saddles are adequately supported by rests. In RaPiD, once the user has selected the connector-drawing tool and clicked on a point within a dental arch to commence drawing a connector, a check is made of all the saddles in the arch. If any saddle is inadequately supported, the user is immediately informed, and is required to remedy the deficient support.

Example (i) above follows a somewhat different pattern from (ii) and (iii). In (i), the user indicates a chosen point for placing a rest, and the design check comes into operation. In examples (ii) and (iii), on the other hand, the user's action is more complex and amounts to the completion of a process: in (ii) this is the drawing of a line linking the end-points of a gingivally-approaching clasp, in (iii) the drawing of the outline of a connector. In both (ii) and (iii) a user, having selected a tool, may immediately misuse it in such a way that it would be pointless to continue. RaPiD's intervention does not, then, have to wait for the user to complete the alteration: rather, the operation fails and a critique is issued straight away.

The two types of critiquing strategy considered so far are, then:
(1) *A critique is issued only when the user has completed the proposed alteration*; and
(2) *A critique is issued immediately upon the user's radical misuse of a tool.*
Three additional modes of critiquing can be usefully employed:
(3) *Critiquing dynamically without negotiation with the user.* This amounts to correcting the user's action automatically and dynamically. The user is not given any choice in the matter, nor informed that the attempted modification is being continuously corrected --- although this fact will be obvious from what happens (or fails to happen) in the graphic window. An example concerns the representation in RaPiD of a tooth's drifting along a dental arch following extraction of a neighbouring tooth. There is a tooth-drifting tool which can be employed to drag the icon of a tooth to a new location. But the tooth is movable only along the line of the arch of teeth.
(4) *Critiquing requested by the user upon completion of a design session, or at certain other stages in the design process.* Certain design rules, concerned with the overall balance of a design, are applicable only at the end of a design session, or at any rate when the user has finished placing all the major components of a denture. There are, for instance, rules determining whether a design ensures the retention of a denture during function. These rules involve some complicated geometrical reasoning, centred on the relation of the denture's axis of rotation to

the retaining elements. RaPiD will apply these rules only when the user has developed the design sufficiently.
(5) *Optional critiquing, requested by the user who wishes to compare his design with that which would have been produced independently by the system.* A user may complete a design without contravening any design rule, but it is unlikely that the design will be *identical* to that which the system would produce if operating in automatic mode. The user may, therefore, wish to compare his own design with one produced automatically. The sort of optional critiquing suggested here amounts to listing differences between the two designs. The user's design can then be brought into line with that of RaPiD, if desired. (The fourth and fifth critiquing strategies have yet to be incorporated in the system; their addition will follow development of the automatic mode.)

Some examples of the different types of critiquing strategy are set out in Table 1 below. Note that the nature of a critique varies according to the type of error committed. Sometimes the fact that a mistake has been made will be so clear to the user that a simple beep will suffice. At other times a detailed message is necessary. When the third critiquing strategy (automatic, dynamic correction) is being followed, this will be obvious simply from the fact that what the user is trying to bring about fails to tally with what actually happens.

Alteration to design	*Constraint to be applied*	*Critiquing strategy*	*Nature of critique*
Placing rest on tooth	1. Rest is in correct mesial/distal position	1	Detailed message.
	2. Tooth must not already carry a rest in the same position.	1	Beep.
Moving tooth to new position	3. Tooth may be moved only to a point on the arch.	3	No message; correction is automatic and continuous.
Drawing a major connector	4. All saddles in arch are properly supported.	2	Detailed message.
	5. Connector does not cover "forbidden" areas (tongue in lower arch, soft palate in upper).	1	Detailed message.
Testing quality of design so far	6. All design rules appropriate for stage reached are adhered to.	4 and/or 5	Detailed message (or sequence of messages).
Positioning a clasp	7. Tooth is natural.	1	Beep.

Table 1 Some examples of the different kinds of critiquing strategies.

6. The design representation employed in RaPiD

Following a recognised convention, a tooth is identified in terms of the quadrant to which it belongs (quadrants 1 and 2 constituting the upper arch of teeth and quadrants 3 and 4 constituting the lower), and its number (in the range 1-8) within its quadrant. A given tooth may have the status *natural* or *artificial* or *missing*. In RaPiD, the components of a design are represented as a logic database, and the statuses of teeth, in particular, are represented by facts of the general form "tooth_in_db(tooth(Quadrant,Number),Tooth,Status)" --- where the variable *Tooth* is instantiated by the name of the graphical object representing a tooth. For example:

```
tooth_in_db(tooth(1,8),tooth_18,missing)).
tooth_in_db(tooth(1,7),tooth_17,artificial)).
```

The existence of components is represented in the database by facts such as "rest_in_db(rest1)", "saddle_in_db(saddle1,[tooth(1,5),tooth(1,6)])", "connector_in_db(upper,connector2,multibar,metal)". These facts state, respectively, that rest1 is a rest, that saddle1 is a saddle carrying artificial teeth 5 and 6 in quadrant 1, and that connector2 is a metal multi-bar connector in the upper arch. There are also facts recording the relationships of components to one another: for example, "clasp_retains_saddle(clasp1,saddle3)".

A user's manipulation of the iconic representation of a design is interpreted as an alteration or update of the design database. If, for example, the incisor tooth (1,1) is to be represented as artificial, the fact "tooth_in_db(tooth(1,1),tooth_11,natural)" is removed from the database and is replaced by the fact "tooth_in_db(tooth(1,1),tooth_11,artificial)". Other design alterations present a more complex picture. If, for instance, a user attempts to place a rest on a tooth, the system surveys the state of the database and determines whether a rest in that position would have the function of supporting a saddle, or some other function such as ensuring the overall stability of the denture. In either case a fact of the form "rest_on_tooth_in_db(Rest,Tooth)" will be added to the database, whereas in the first case only, an additional fact of the form "rest_supports_saddle_in_db(Rest,Tooth,Saddle)" will also be added.

A RaPiD design is in the form of a deductive database and can be thought of as a set of clauses comprising both the facts relating to a given case and the design rules applicable to any case. Facts about the individual case are always atoms, while rules are expressed as extended Horn clauses (Kowalski, 1979), of the form

$$A \leftarrow W$$

where A is an atom and W is any expression (atomic, negative or conjunctive) of first-order logic. We could, for example, express as follows the rule that a rest supporting a free-end saddle should be placed on a tooth mesially:

```
rest_on_tooth(Rest,Tooth,mesial) ←
    saddle_in_db(Saddle,Artificial_teeth_on_saddle),
    tooth_associated_with_saddle(Tooth,Artificial_teeth_on_saddle),
    type_of_saddle(Saddle,Artificial_teeth_on_saddle,free_end).
```

This deductive-database approach provides (1) a clear, unambiguous description of the components in a design and of their relationship to one another, (2) improved modularity for grouping design rules, and (3) a logically sound procedure for checking database updates. (For general theoretical accounts of the nature of a deductive database, see Sergot, 1991 and Das, 1992.) To test whether a design modification is acceptable, we check the proposed change against the design rules, expressed as integrity constraints, conditions which the database is required to satisfy. Many of the integrity constraints used in RaPiD are dynamic, that is, constraints governing the transition from one state of the database to another during an attempted update. As such they are applied as *preconditions* for allowing a proposed modification to succeed. If an attempted modification satisfies all its preconditions, it is accepted and the database is updated. If the input fails to satisfy even one of its preconditions it is rejected and the database remains as it was. For instance, it is a precondition for placing a rest on a tooth that the tooth is a natural one; it is a precondition of placing a saddle on an edentulous area that that area is not already covered by a saddle. Given a full range of *precondition* clauses, encompassing all the various types of input, we rely on a procedure *assimilate* in the style of Kowalski and Sergot (Kowalski, 1979; Sergot, 1991) to handle the incorporation of attempted alterations. If the preconditions for an alteration are satisfied, an *update* procedure is used to incorporate the alteration in the database; if one of the preconditions is not satisfied, the alteration will not succeed and the nature of the failed precondition will determine whether a message should be presented to the user, or whether there should be some other response such as a beep.

7. Evaluation of the system

RaPiD's user-interface and its graphical capabilities have been extensively tested by the project team and also evaluated by a consultant dental prosthetist, a panel of dental practitioners and a number of dental technicians. Their comments have been recorded and relevant improvements incorporated into the software. This process of evaluation by users of RaPiD as a graphical tool is continuing and will recruit teachers and students in other UK dental schools and postgraduate centres. In addition, an assessment of the graphical quality of the designs produced by RaPiD has been completed. A series of designs produced manually by students under staff supervision was obtained. Each design was reproduced using RaPiD, and both versions were returned together with a questionnaire to the originating dental staff for comparative assessment. The results for the RaPiD versions of the design were highly favourable (Hammond et al, 1993b).

The design rules which are being incorporated into RaPiD are based, not on local preference, but on an extensive survey (now complete) of the literature. A detailed questionnaire has also been circulated to prosthetic specialists at all UK dental schools, aimed at gauging levels of acceptance of various design rules. The responses to this questionnaire of 10 specialists at the School of Dentistry in Birmingham have been recorded and analysed. The questionnaire is to be distributed to co-operating schools abroad.

An assessment of the educational effectiveness of RaPiD is planned, and will be undertaken in two stages:

(a) A project, undertaken in co-operation with the Faculty of Education, University of Birmingham, to measure any improvement in students' grasp of the design rules, and their ability to design a denture in accordance with the rules, resulting from the use of RaPiD. The project involves working with two groups of students in the Dental School who have completed their basic course in RPD design. One group will use a version ("expert" mode) of RaPiD in which the design rules are not applied to restrict users' actions in any way, although their achievement will be recorded through an error log of design rules contravened during the design process and by a design quality index of the finished design: the latter index is a measure of the compliance of the finished design with the established design rules. The other group will use RaPiD in "student" mode with design rules operative so that students are guided to an optimal solution; again an error log and quality design index will be obtained.

(b) 100 consecutive hand-drawn RPD designs for patients, produced by students without RaPiD's assistance, have been collected. The compliance of these designs with the design rules is to be measured to produce a design quality index. The full "student" version of the system, once it has been completed, will be used to produce all designs by students. A post-RaPiD design quality index will then be obtained and compared with the pre-RaPiD index to indicate the level of improvement within the school as a whole achieved by introducing the system.

8. Related work

Systems for RPD design have been described by Maeda et al. (1985 and 1987) Wicks and Pennell (1990) and Beaumont (Beaumont & Bianco, 1989; Beaumont, 1989). For some critical remarks about these systems, see Hammond et al, 1993b. The *Kontest* system (Jakstat et al, 1991) is concerned with undergraduate education; its effectiveness is, however, restricted because much data must at present be input at the keyboard. It has not been possible to appraise the *Stelligraphe* system (J. Gaillard, Appolline Productions, Sainte-Usage, Louhans, France), now in commercial use, because public-domain descriptions of the software are not yet available.

9. Future work; concluding remarks

Expanding the range of design rules in RaPiD so that it becomes comprehensive is a current priority, as is the introduction of critiquing strategies (4) and (5) (see section 5 above), as well as completion of the automatic mode of operation. The programme of evaluation of all aspects of RaPiD is now being actively pursued. The extensive use and testing of RaPiD already carried out show it to be a versatile and robust knowledge-based system whose benefits in dental education will be widespread.

Acknowledgement FJF and DAR are funded under the KBS Initiative of the UK Higher Education Funding Councils.

References

Basker, R. M. & Davenport, J. C. (1978), A survey of partial denture design in general dental practice, *Journal of Oral Rehabilitation* 5, 215-222.

Basker, R. M., Davenport, J. C. & Harrison,A. (1991), Designing removable partial dentures in general dental practice, Proceedings of 38th Annual Conference of the British Society for the Study of Prosthetic Dentistry.

Beaumont, A. J. & Bianco, H. J. (1989), Micro-computer removable partial denture design, *Journal of Prosthetic Dentistry* 6(2), 417-421.

Beaumont, A. J. (1989), Micro-computer removable partial denture design: the next evolution, *Journal of Prosthetic Dentistry* 6(2), 551-556.

Das, S. K. (1992), *Deductive databases & logic programming*. Reading, USA: Addison-Wesley.

Hammond, P., Davenport, J. C. & Potts, A. J. C. (1993), Knowledge-based design of removable partial dentures using direct manipulation and critiquing, *Journal of Oral Rehabilitation* 20, 115-123.

Hammond, P., Davenport, J. C. & Fitzpatrick, F. J. (1993), Logic-based integrity constraints and the design of dental prostheses, *Artificial Intelligence in Medicine* , 5, 431-446.

Holt, R. D.,Rule, D. C., Basker, R. M., Davenport, J. C., Ralph, J. P., Murray, J. J. & Eaton, K. A. (1994), The influence on partial denture design of a teaching video for general dental practitioners, *British Dental Journal* , in press.

Jakstat, H., Dreschler-Fischer, L., & Burmester, V. (1991), Vorstellung eines Expertensystems zur Planung von partiellem Zahnersatz, *Deutscher Zahnärztliche Zeitschrift*, 46, 226-228.

Kowalski, R. A. (1979), *Logic for Problem Solving*. New York: North-Holland.

Maeda, Y., Tsutumi, S., Minoura, M.,Okada, M., Nokubi, T.& Okuno, Y. (1985). An expert system for designing removable partial dentures - preliminary report, *Journal of the Osaka University Dental School* 25, 79-84.

Maeda, Y, Tsutumi, S.,Okada, M., Minoura, M.,Nokubi, T.& Okuno, Y. (1987), An expert system for designing removable partial dentures - the role of database, *Journal of the Osaka University Dental School* 27, 75-82.

Sergot, M. J. (1991), Lecture notes on logic databases, Department of Computing, Imperial College, London.

Wicks, R. R. A. & Pennell, M. E. (1990), A computer-assisted design guide for removable partial denture frameworks, *Dental Practice* 29 (6) 14.

Silent Movies: A Digitized Video Approach to the Russian Verb

LISA A. FRUMKES
Slavic Languages & Literatures
University of Washington, DP-32, Seattle, WA 98195, USA
E-Mail: frumkes@u.washington.edu

Abstract: A HyperCard stack using digitized video to provide students of Russian with meaningful exercise in the use of verbal aspect and verbs of motion and position is described. Examples of traditional textbook exercises are presented and shown to provide inadequate practice in the use of these verbs. A new alternative is suggested: video without an audio track gives students pedagogically sound practice by allowing them to independently generate full sentences to describe what is happening in a scene, while making only minimal reference to their native language. Special aspects of the program, such as the variety and flexibility the of feedback available to students, are also discussed.

"Silent Movies" is a context-based program which gives students of Russian practice using verbal aspect and verbs of motion and position. The student's task is to write full, grammatically accurate sentences in the target language to describe the images she sees. This sort of program fulfills the requirements for meaning-enhancing drills set forth by Chun and Brandl (1992). The video-without-audio approach was implemented because traditional textbook approaches to the Russian verb fall short: they often provide inadequate context for students to make a good judgement about which verb to use, encourage students to rely on problem-solving techniques which are of little use in real conversations, or indirectly cause confusion about the nature of the verbs. By contrast, "Silent Movies" encourages the student to use the target language without constant reference to the native language or to clues from the target language. A simple parsing mechanism analyses student input and pinpoints grammatical and conceptual errors therein. The student is provided with hints which will aid her in correcting her answer. In addition, the student is in control of this feedback, choosing the type and amount of it she sees.

In the past, video has been viewed almost exclusively as a tool to exercise or test listening comprehension. Students usually use such exercises by choosing answers from a multiple-choice list or filling in blanks. But images, without audio, are a valuable form of communication which can be used to prompt students into independently generating full sentences. It is this potential of video that "Silent Movies" exploits. In addition, the program demands very little in terms of computer resources: it requires only a Macintosh IIci (or better) computer with a hard drive, HyperCard and QuickTime.

Problems Presented by the Russian Verb

In order to understand how "Silent Movies" can benefit students in a way that traditional textbook and classroom exercises cannot, we must first understand the complexities of the Russian verb system. Below, I briefly describe the three types of verb oppositions which "Silent Movies" exercises: aspectual (imperfective/perfective), directionality (unidirectional/multidirectional), and movement (active/static).

Verbal aspect in Russian distinguishes between actions in process (imperfective) and completed actions (perfective) in the past and future tenses. The majority of Russian verbs are paired for aspect. Some examples of aspectual pairs: imperfective *chitat'* 'perform the action of reading' and perfective *prochitat'* 'read completely'; imperfective *pit'* 'perform the action of drinking' and perfective *vypit'* 'drink up, drink it all'.

When using verbs of motion, one must determine whether a unidirectional verb (indicating motion in one direction or motion occurring at the moment of speech) or a multidirectional verb (expressing motion with no particular goal, or a complete or round trip) is appropriate in a given situation. Examples of such unidirectional and multidirectional pairs are *idti* and *xodit'* 'go by foot', *exat'* and *ezdit'* 'go by vehicle', *nesti* and *nosit'* 'carry', *vezti* and *vozit'* 'convey by vehicle'.

Verbs of position show both imperfective/perfective pairing as well as active/static oppositions. For example, the active aspectual pair *lozhits'a/lech'* 'lie down' contrasts with the static aspectual pair *lezhat'/polezhat'* 'be in a lying position', just as the active aspectual pair *vstavat'/vstat'* 'stand up' is parallel to the static aspectual pair *stojat'/postojat'* 'be in a standing position'.

Traditional Approaches to the Russian Verb

Verbs are easily taught and exercised in the classroom, where a teacher can add appropriate context with body and object movement to facilitate student comprehension. Written homework, usually in the form of translation or fill-in exercises, however, cannot provide such easily understood context. Instead, it must provide context in the target language. Let us examine these textbook approaches in greater detail.

Translation

The translation approach is one of the most familiar and most criticized of exercise tools. The native language sentences are often stilted to hint at the proper translations into the target language. Examples (1) and (2) below come from Russian elementary and intermediate textbooks:

(1) "Olga went to the store, there bought a skirt, blouse and shoes and after that went to a movie." (Clark, 1983)
(2) "Today we studied physics and chemistry. We worked on problems all day, but we did not solve all the problems." (Davis and Oprendek, 1973)

Fill-ins

In the fill-in exercise, students see sentences from which verbs are missing. Their task is to fill in the proper verb based on their understanding of one or two sentences of surrounding context. Cues are provided to help students fill in the correct word. Such cues are of two types: (1) a native language gloss of the desired word or (2) a choice of words in the target language. Target language cues may be in a dictionary form (in the case of verbs, the infinitive) or may already be properly inflected. Examples (3) through (6) illustrate this sort of exercise.

Native language cues:
(3) Ja obychno (go to bed) spat' rano, no vchera ja (went to bed) ochen' pozdno. (Davis & Oprendek, 1973)
(3') 'I usually go to bed early, but yesterday I went to bed very l[illegible].'
(4) Volodja (took off) shapku i sel za stol. (Clark, 1983)
(4') 'Volodya took off his fur cap and sat down at the table.'

Target language cues:
(5) Kuda èto rabochije (nesti/tashchit') rojal'? (Muravyova, 1986)
(5') 'Where are those workmen (carry, infinitive/lug, infinitive) that piano?'
(6) On segodnja (zabyval/zabyl) svoi knigi. (Davis & Oprendek, 1973)
(6') 'He (forgot, imperfective/forgot, perfective) his books today.'

Other fill-in exercises provide context in the form of an entire paragraph in the target language and may also use cues of the types described above. In example (7), the student is instructed to insert appropriately prefixed forms of the verb *idti* 'go, walk' into the blanks so that the entire paragraph will make sense.

(7) Vchera my reshili _____ v kino. My _____ iz doma i _____ po ulice. Kogda my _____ po ulice, nachalsja dozhd'. My reshili _____ v magazin i nemnogo podozhdat'. Dozhd' skoro konchilsja, my _____ iz magazina i _____ dal'she. Cherez neskol'ko minut my _____ v kinoteatr. My kupili bilety i _____ v zal. (Davis & Oprendek, 1973)
(7') 'Yesterday we decided to go to the movies. We exited the house and set [illegible] along the street. While we were walking along the street, it began to rain. We decided to enter a [illegible] and wait a little while. Soon the rain stopped, we exited the store and set off further. After a few minutes we arrived at the theater. We bought tickets and entered the auditorium.'

Picture-based Approaches

An approach where students describe the action depicted in static pictures can be used quite successfully in the classroom. Unfortunately, students working on them at home without guidance may not learn very much, since they will not see or understand their mistakes until they speak directly with an instructor. An example of such a picture-based exercise is found in Figure 1, below.

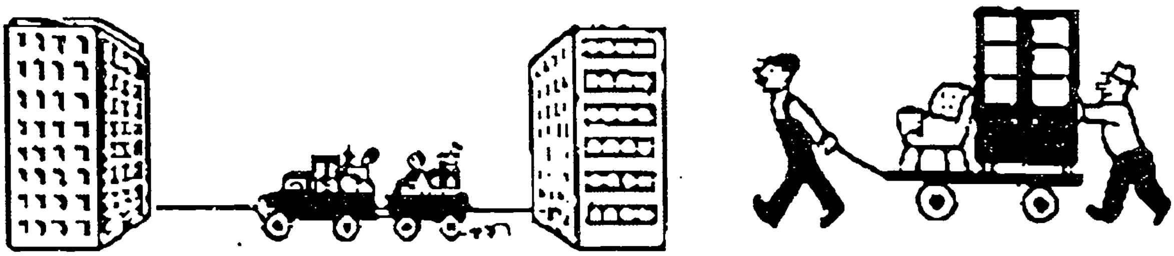


Figure 1. An example of a picture-based exercise. (Muravyova, 1983.)

Improving on the Traditional Approaches Using the Computer

We must find a better way for the student to get sufficient practice in the use of these verbs outside the classroom. I suggest that students need visual context and guided error correction to fully understand and use these forms properly. The computer coupled with digitized video is ideal for providing this opportunity.

Picture-based approaches are an improvement over translation and fill-in exercises for several reasons. First, they do not encourage students to translate from the native language into the target language. This is valuable, since successful language learners learn to think in the target language without going through the intermediary of the native language. Second, they avoid the artificiality of target-language clues which force the student to continually refer to a dictionary form as the basis for choosing a correct inflected form.

The approaches outlined above are still the best a paper textbook can provide for independent practice. But even when textbook exercises provide good and copious context, individual students may still perceive situations described in written narratives in a manner which the individual checking the homework did not expect. As a result, an answer is marked wrong because it does not agree with the instructor's interpretation of the context, although it may be correct for the student's interpretation of it. While such misunderstandings can be resolved immediately during in-class drills, written homework entails a time-lag; by the time students receive corrected homework back, they may not recall the rationales for the answers they gave, and thus gain very little from the corrections. And while it is obvious to instructors which verb form the context is aiming for (since they have so much experience with such exercises) it is often not at all obvious to the first- or second-year student. In other words, the student is forced to learn to read the contexts in an expected way instead of learning to use the verbs correctly. Furthermore, since this context is written in the target language, the student who is already struggling with the language will become further frustrated.

Fill-in and translation exercises generally give a clue about the form the learner needs to use. Students learn to look for key words to determine which forms to use in various contexts and come to rely on these clues. For example, students learn that if they see a word relating to the frequency of an action (such as *obychno* 'usually', *chasto* 'often', *redko* 'rarely' or *nikogda* 'never') that they need to use the imperfective aspect of the verb to indicate a repeated or habitual action.

(8) Misha *chasto* (missed) lekcija, no Masha *nikogda* ne (missed) (Clark, 1983 (italics mine))
(8') 'Misha *often* missed classes, but Masha *never* did (miss classes).'
(9) On *vsegda* (sprashivat'/sprosit') menja. (Davis & Oprendek, 1973 (italics mine))
(9') 'He *always* asks me.'

Another method students commonly use to guide them to the correct form is reference to a verb before or after the fill-in. For example, students learn that if actions are sequenced in time that they both take perfective verbs. Therefore, if the student understands the sentence well enough to see that the actions are sequenced, and that one verb in the sentence is in the perfective, she knows to use the perfective for the fill-in verb as well.

(10) My *soshli* s tramvaja i (transferred) na avtobus. (Clark, 1986 (italics mine))
(10') 'We *got off* (perfective) the tram and transferred onto a bus.'
(11) Reb'onok (vkljuchal/vkljuchil) televizor, i *nachal* smotret'. (Davis & Oprendek, 1973 (italics mine))

(11') 'The child (turned on, imperfective/turned on, perfective) the television and *began (perfective)* to watch.

While these sorts of devices help students to *achieve* well on exams, there is no guarantee that it aids their *proficiency* in the language.

How "Silent Movies" Works

What the Student Sees and Does

In this section I will briefly describe how "Silent Movies" works. Figure 2 shows the approximate appearance of the exercise screen to the student.

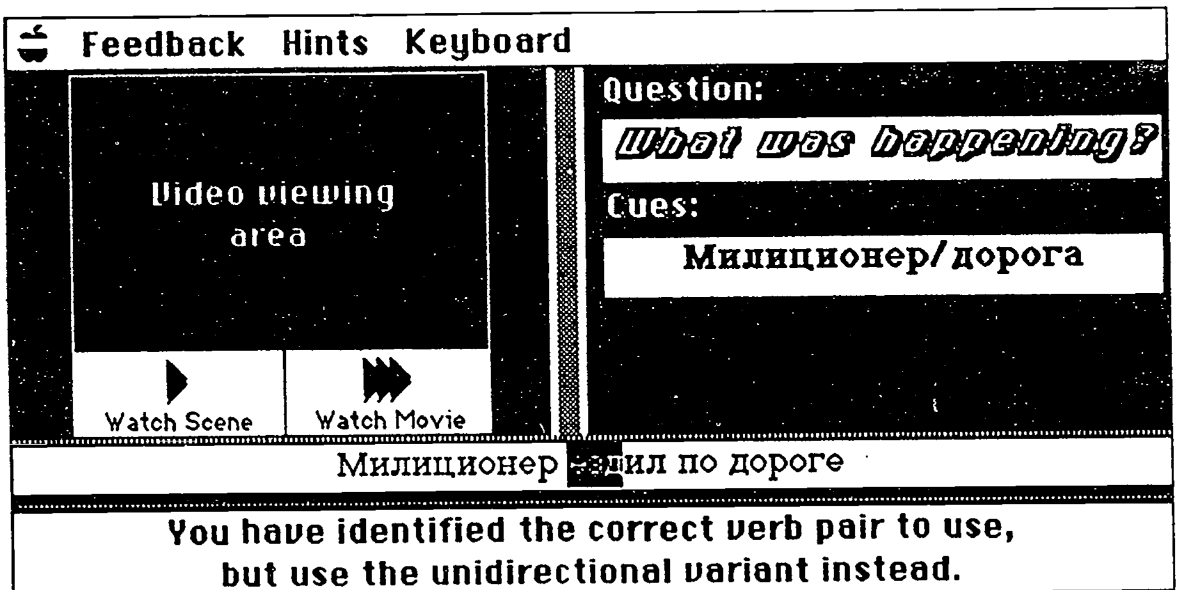


Figure 2. The "Silent Movies" exercise screen (reduced and modified due to space considerations).

The student watches a mini-movie which lasts no longer than one minute. This gives her a sense of the story she will tell in her descriptive sentences. She may watch the entire movie whenever she likes throughout the exercise, but after she has watched it once, she is invited to watch the first of several scenes from the mini-movie. Similarly, the student may watch individual scenes as many times as she desires, but after the first showing, her attention is drawn towards a question of the type "What was happening?" or "What happened?" The question indicates the tense the student will use in her sentence; it is given in English so that it is not obvious which verbal aspect is necessary. In addition, the student sees the nouns and adjectives she will use in her sentence. This narrows the number of possible answers. In the mini-movie "Lost Dog", for example, the first scene shows a policeman in a car driving down the road. Possible descriptions of this scene might be "The policeman was in the car" or "The policeman was driving along the road" or "Officer Ivanov was driving down the road in his car." Since the cues given are 'Milicioner/doroga' (policeman/road) and the question is "What was happening?", the student is directed towards the answer "The policeman was driving along the road."

When typing in her answer, the student has several tasks to fulfill. The first and most important is selecting a verb to describe the action being performed, based on the context provided by the video. Of secondary importance is using the correct tense, person, gender and number in its conjugation and providing necessary prepositions and correct endings for adjectives or nouns. Thus, grammatical accuracy is required only after the communicative task of selecting the correct verb has been successfully completed.

The "Parser"

In order to allow the student to focus on meaning first and grammatical accuracy second, a certain amount of parsing is necessary. Each problem in the database contains several "codes" which provide information to the program about the correct answer. One code allows the program to identify the category (noun, verb, adjective, preposition, etc.) of each word in the sentence, a second one indicates the cases (nominative, accusative, dative, etc.) of individual nouns and the tenses and aspects of the verbs, while a third code divides each word into stem and ending. The program can then use this information to isolate whether the student's error lies in the stem or ending and give hints to aid the student in correcting the problem.

Feedback

Research has shown that students appear to learn best when they have some control over the feedback they receive (Robinson, 1989). In "Silent Movies," the student controls when she gets feedback and how detailed that feedback is. Two sorts of feedback are found in "Silent Movies": error-correction and grammatical aid feedback.

Error-correction Feedback

"Silent Movies" provides three types of error-correction. The first type is a simple right/wrong response; this is the default, provided automatically when the student presses <return> after typing in an answer. The second level of error-correction highlights the first error found, whether it is an entire word or just a stem or an ending. The last type highlights the error and gives a hint about how to correct it. All three types of feedback are available to the student on demand after she has given one incorrect answer.

Grammatical Aid Feedback

As the student enters additional uniquely incorrect guesses, more explicit feedback becomes available to her. This feedback consists primarily of grammatical and lexical help.

The first type of grammatical aid feedback is a clarification in English of what is happening in the picture. It becomes available to the student after the first incorrect answer has been entered. There are two reasons why students might need this sort of help. First, since "Silent Movies" is not geared for use with any particular textbook, students may be unfamiliar with some of the vocabulary used in the cues. The English translation provides a contextualized gloss for these words. Second, sometimes the action in the video can be interpreted in more than one way. For example, in the "Lost Dog" mini-movie from "Silent Movies", we see a dog in the foreground, while a police car pulls up and stops on the road behind the animal. This may present the student with a quandary: should she describe the position of the dog, or the motion of the car, or the motion of the car in relation to the position of the dog? The English translation can immediately clear up such ambiguities.

After two incorrect answers have been given, the student may choose to see which verb pair (imperfective/perfective, unidirectional/multidirectional, active/static) she must choose from. For example, if the video clip shows the policeman in the act of crossing a street, she must choose between the imperfective verb *perexodit'* or the perfective verb *perejti* to describe the motion.

Also available after two incorrect answers is feedback which tells the student which verb tense is required. In theory, the student should not need this help, if she has read the question carefully, but experience shows that students often forget what tense they are dealing with when they are concentrating on aspect and directionality.

After three incorrect answers, the student is given access to aspect or directionality help; that is, she can choose to be told whether to use a perfective or imperfective, unidirectional or multidirectional, active or static verb. So in the example given above, where the policeman is crossing the street, the student would be told that the imperfective choice (*perexodit'*) was appropriate.

After four incorrect answers, the student can find out which prepositions the sentence requires, as well as what case forms the nouns and adjectives take.

Finally, after the student has made five attempts to discover the correct answer, she has the option of giving up, seeing the correct answer and going on to the next problem. It should be emphasized that students are not obliged to use any help at all--but if they want it, it is under their control.

Miscellaneous Features of "Silent Movies"

All student responses are collected and stored and the student is given the opportunity to review her incorrect answers at the end of the exercise. These responses may be reviewed by instructors as well, thereby enabling them to address common problems in class. In this way, the instructor does less homework correction, but still gets insight into how well students are learning and understanding.

The scoring used in "Silent Movies" is forgiving; if the student finally comes up with the right answer, she gets full credit no matter how many times she answered it incorrectly. These scores are easily collected, divided into course sections and distributed to instructors.

The Future of "Silent Movies"

Initial informal testing of "Silent Movies" with second-year Russian students was encouraging. Strong students said they felt challenged, and weaker students were allowed to learn at their own pace and did not become discouraged or give up as soon as allowed. Formal tests will continue in the future and will include control groups working with traditional textbook approaches.

Not yet implemented, but arguably a good addition to "Silent Movies", would be a grammar reference. Such an addition would be a good project for a graduate or advanced graduate student of Russian who has minimal computer experience as a good introduction to the use of HyperCard.

"Silent Movies" could be improved in one important way. The video it currently uses was taken from satellite television news broadcasts; as a result, only a limited selection of video clips was available to me as I developed the program. Ideally, scripted filming on location in Russia with actors would be done in order to concentrate on the most common problems students experience with Russian verbs. For instance, one would like to see negated actions (which take the imperfective) illustrated by scenes such as the following: a man enters a room, opens a window, takes a look out, and closes the window again (*On otkryval okno*). Such scenes would be contrasted with non-negated ones (taking the perfective) such this one: a man enters the room, opens the window, and leaves again without closing it (*On otkryl okno*). Scripted filming would also allow us to eliminate action occurring in the background which can distract the student from concentrating on the material relevant to the exercise.

The next stage of "Silent Movies" will involve designing a simple interface allowing instructors of Russian to use their own video clips and design exercises around them. It is hoped that funding will become available for this project in the near future.

References

Chun, D. E. & Brandl, K. (1992). Beyond Form-Based Drill and Practice: Meaning-Enhancing CALL on the Macintosh. *Foreign Language Annals*, 25(3), 255-267.

Clark, B. T. (1983) *Russian*. New York: Harper & Row.

Davis, P. A. & Oprendek, D. V. (1973) *Making Progress In Russian*. New York: John Wiley & Sons.

Muravyova, L. (1986) *Verbs of Motion in Russian*. Moscow: Russky Yazyk.

Robinson, G. L. (1989) The CLCCS CALL Study: Methods, Error Feedback, Attitudes, and Achievement. Wm. Flint Smith (Ed.) *Modern Technology in Foreign Language Education: applications and projects*. Lincolnwood, IL: National Textbook Company.

Acknowledgements

I would like to thank the Language Learning Center at the University of Washington for its financial and moral support of this project, and Paul Aoki and Klaus Brandl, as well as two anonymous reviewers, for their comments on this paper.

Video conferencing with preschool children: Mass communications media in music instruction

PETER GOUZOUASIS
Department of Visual and Performing Arts in Education
University of British Columbia, 2125 Main Mall, Vancouver, BC, V6T 1Z4, Canada
E-mail: petergou@unixg.ubc.ca

Abstract: An experiment that involved video conferencing technology with preschool children in a music instruction context. Videoconferencing is a powerful communications medium, and may be used in creative interactive contexts. A detailed discussion of rehearsal procedures, technical broadcast information, instruction techniques, evaluation of instructional efficacy with objective tools, and interpretations of the data is provided. Whereas children's television programming—especially in a music context—is essentially non-interactive, video conferencing provides educators a viable, interactive audio-video medium to deliver instruction in a variety of subject areas.

Introduction, purpose, and problems

Mass media are globally pervasive. We live in a global village (McLuhan & McLuhan, 1988) where through a massive network of satellites, telephone cables, optical cables, and microwaves electronic media have linked nations across continents. In most communications situations, the speed of the flow of information in the Digital Age outpaces our ability to assimilate, accommodate, and synthesize new information. As McLuhan (1964) hypothesized, new technologies constantly evolve, and in many cases envelope existing technologies. As a contemporary example, RISC processor technology, seems poised to replace CISC technology in desktop computers. The majority of people who participate in the work force are not "technoliterate," and many seem oblivious to the notion that developments of the past decade in communications technology have led to a distinct change in the way humans process information. Even though we are in the midst of an information explosion of magnanimous proportions, education ministers, education policy makers, and politicians seem oblivious to the numerous technological developments that are emerging as we approach the 21st century.

Media define culture (Gouzouasis, 1993). All that we learn is assimilated through some communications medium—the human voice, print media, audio recordings, video recordings, radio, television, and computers. Indisputably, mass media—in the form of radio, television, motion pictures, telephones, and computer controlled information systems—are pervasive. Essentially, we are what we assimilate through media. In this day and age, instruction may be delivered both purposively and indirectly, and learning may take place both consciously and unconsciously (i.e., subliminally). In a contemporary example, whereas the much maligned "Barney" children's television program is an example of purposive use of music material and conscious music making, "Sesame Street" is an example of how music can be relegated to background "noise" and a visual support role. In general, television programming and commercials use music as background information, usually as a sound effect, rather than foreground information. Inevitably, in multimedia, as in most hot media, visual information takes precedence over acoustic information (i.e., music). Subsequently, we seem to live in a visually stimulated, musically-illiterate society.

Videophones and video conferencing systems are a accessible reality. A videophone is a telephone with screen attached to its apparatus. Video conferencing technology involves the use of both video and telephone technologies. In the typical video conference, people use the audio to lecture and discuss issues, and the visuals are used to see participants and project a variety of diagrams and illustrations. Bell Atlantic is poised to promote videophones on a mass marketing level and currently sells them for approximately $2500. Recent mergers of large communications corporations will forever change the way that we communicate, both in everyday life and in educational settings.

Burtenshaw (1992, 1993a, 1993b) is a pioneer in the use of video conferencing technology in education. His experiments involve the successful use of video conferencing technology in a variety of music teaching contexts. Teaching segments in various instruments (oboe, violin, and trumpet), piano master classes, and a one-way early childhood lecture-demonstration comprised his earliest projects (Burtenshaw, 1992). In the piano master classes, students were critiqued and given performance directions by master teachers, but the instructional techniques were limited range and flexibility, primarily because of the lack of familiarity (on the part of master teachers) with video conferencing's interactive potential. Rather than fully-interactive sessions, where participants are able to immediately respond in an echoic (rote) or sequential manner to a teacher's instructions, most video conferencing experiments involve the observation of teacher-student interactions in live teaching situations (MacKinnon & Scarr, 1992), lectures, and discussion sessions. It was from the perspective of expanding this relatively new medium to its ultimate application potential that the present study was undertaken.

With the intent of exploring mass media in education, the purpose of this study was to learn more about the use of video conferencing in a music learning environment. Specifically, the the primary problem was to examine the efficacy of video conferencing in a rhythm instruction segment. Concomitant with the purpose, because no objective evaluation criteria have been developed for video conference sessions, the secondary problem was to develop objective evaluation techniques for video conferencing instructional sessions.

Methodology

Samples and sites

Eight children, ages 3 through 5, participated in the video conference broadcast from Australia. The children comprised an availability sample, and were included in the experiment based on their ability to attend the video conferencing studio in Sydney, Australia. The researcher received a list of the children's names by fax and randomly assigned each child's name to a two beat pattern on a list of eight patterns. Eight children, ages 3 through 5, participated in the video conference broadcast from Canada. The children were preschool students of the researcher, and were included in the experiment solely based on their ability to attend the video conferencing studio in downtown Vancouver, British Columbia. Two days before the video conference, the researcher faxed a list of the Vancouver children's names to the music specialist in Sydney, Australia. Names were exchanged in this manner to facilitate instruction and to help make the children more familiar during the instructional segments. Also, each child on each continent wore large name tags to facilitate instruction. The session, which took place in the video conferencing studios of The Commonwealth of Learning in downtown Vancouver, was transmitted on Wednesday, May 19, 1993, from 4:00 p.m. to 5:30 p.m. Vancouver time, and received on Thursday, May 20, 1993, from 9:00 a.m. to 10:30 a.m. Sydney time.

Technical Information

The transmission used two standard telephone lines (analog information), converted by BC Telephone Company into 56kb lines (kilobits of digital information). That information was transmitted via CLI Codec (coder-decoder) and translated in Coff's Harbour, Australia into 64kb ISDN (Integrated Services Digital Network) information that was received by the PictureTel Codec in Sydney, Australia. The reason for the multipoint bridging (Vancouver to Coff's Harbour to Sydney) was that North American equipment uses a 56kb line standard, while the international standard is 64kb lines. At the time of this experiment, there were no p X 64 (proprietary 64kb) bridges, hence the translation from 56kb affected our video resolution, video movements, and audio-visual coordination. In essence, whereas the video information was transmitted at both ends in compressed 15 frame per second video, the translation bumped that resolution down to approximately 8 frames per second. In addition to the Vancouver-Sydney link, to allow other art and music educators to view and comment on the experiment, the signal was also transmitted to Armidale, Australia.

General Transmission Procedures

For the Vancouver transmission, two rehearsals were conducted prior to the instructional broadcast. During the first rehearsal which involved a live transmission to Sydney, the researcher determined that movement activities would need to be drastically limited because of the delay and resolution in the video signal. The researcher sang a children's song to the technical engineer in Sydney, and attempted to teach the same song with the rote-phrase technique to the technical engineer. A gesture that is typically used by teachers to cue children to echo sing or chant (i.e., both hands on chest, teacher's turn to sing; both hands extended to children, children's turn to echo sing) in the rote-phrase technique did not seem to be affected by the resolution of the video signal, and it was decided that the gesture could be interpreted as an invitation for the children to respond to the Vancouver-based researcher's chanting of rhythm patterns.

During the second rehearsal, the researcher's entire presentation was timed and the quality of a ten minute instructional video demonstration was evaluated by the video conference educational technician, a cameraman, and research panel participants. The instructional video included four segments: (1) diatonic stepwise tonal pattern singing instruction in major tonality, (2) diatonic arpeggioed tonic and dominant tonal pattern singing instruction in major tonality, (3) four beat rhythm pattern chant instruction in duple meter, and (4) two beat rhythm pattern (performance on hand drum) instruction in duple meter. The researcher hoped that the children in Australia would watch parts of the video during the video conference broadcast and develop a sense of familiarity with the researcher through his work with the children on the videotape. In a sense, I thought it was possible to provide a model for the type of activity in which the children Sydney would participate. Since most children seem to be attracted to information transmitted by television, the researcher believed that viewing the children who performed music on the Vancouver videotape would establish a mood for the children in Sydney to work within this medium.

Instructional procedures: Transmission to Australia

To begin the entire broadcast, the group of children who were in the Vancouver studio sang a "Welcome Song" to the group in Sydney. The educational technician used the remote control touch screen module to turn on the videotape machine to broadcast the instructional demonstration. He muted the microphones on the participant's table, but neglected to mute the overhead studio microphones. That caused an echo effect in the Armidale studio, which made the videotape segment difficult to interpret in Armidale, Australia. At the end of the 10 minute video broadcast, the researcher was seen live on camera in Sydney, Australia. As he greeted each child individually, according to the list provided via fax, the Sydney studio video camera scanned the group. As the researcher gestured with his hands and arms, he chanted "My turn first then your turn." Each line of the rhyme "Pease Porridge Hot". was echo chanted, phrase-by-phrase, with words. The researcher repeated his gesture and repeated the previous directions. Next, each line of the rhyme "Pease Porridge Hot" was echo chanted, phrase-by-phrase, without words on the syllable "ba." "Ba" was used because it is considered an easy syllable to chant for very young children. Moving from a rhyme ("Pease Porridge Hot") with words to the same rhyme chanted with a consonant-vowel syllable ("ba") is deconstructivist in nature. Rhythm exists in speech patterns, but the fundamental nature of rhythm as an element of music is often overshadowed by words.

Immediately following the rhyme chanting activity, each Australian child was asked to echo chant a two beat rhythm pattern. Eight two beat rhythm patterns in duple meter (2 meter beats, quarter note represents the beat) were selected (one for each child) for instruction based on the notions that (1) from a Gestalt perspective, a one measure, two beat rhythm pattern is the smallest logical unit of music information that can impart a sense of meter, and that (2) for young children, two beat rhythm patterns seem easier to perceive and conceptualize, and easier to perform. The ability of Australian preschool children to chant a two beat rhythm pattern on the syllable "ba" was examined to determine interactive efficacy of video conferencing as a basic music teaching tool.

The music specialist in Sydney, Australia introduced herself with a "Hello Song" based on the melody of "The farmer in the dell." Two children in the Vancouver studio immediately recognized the song, and began singing along with the words to "The farmer in the dell." Next, the Australian researcher taught a segment on the concepts of high and low pitch direction through movement in song (i.e., children move entire body down when the pitch sounds low and up when the pitch sounds high). It should be noted that in and of itself, the concept of "direction" in the form of "high" and "low" direction is an adult imposed music construct, and young children are able to sing in tune and discriminate individual pitches and melodic patterns without such directional knowledge.

Evaluation of the transmission and rhythm pattern chanting segment

A questionnaire composed of 16, five-point, rating scales was used to evaluate the efficacy of the video conference transmission and the Vancouver teacher's rhythm pattern instruction (see Appendix A). Four independent raters—two music specialists who sat separately in the Vancouver (R1 and R2), the researcher (R3)(Vancouver), and a music specialist based in Sydney (R4)—were enlisted to judge the video conferencing session. A section for anecdotal comments was also provided to gather each rater's overall observations and general perceptions about the technology.

Results

Reliabilities of the rating scales

The interrater reliabilities for the pairwise rater comparisons are presented below (see Table 1).

Table 1
Interrater Reliability Coefficients

	Rater 1	Rater 2	Rater 3	Rater 4
Rater 1 (R1)	----	----	----	----
Rater 2 (R2)	.704	----	----	----
Rater 3 (R3)	.573	.710	----	----
Rater 4 (R4)	.411	.499	.156	----

Generally, it can be seen that the level of agreement (r^2) between the Vancouver based raters, in terms of a perspective of what occurred during the video conference, was higher (R1-R2=49.5%; R1-R3= 32.8%; R2-R3= 50.4%) than the level of agreement between the Vancouver based raters and Australian rater (R1-R4= 16.8%; R2-R4= 24.9%; R3-R4= 2.4%).

The efficacy of the teaching segments

As was demonstrated on the instructional videotape, it seems that children are able to focus on and perform rhythm patterns when they use simple, consonant-vowel (cv) syllable combinations ("ba"). Furthermore, with one exception, all of the children (8) in Sydney were able to accurately chant a rhythm pattern. Through the anecdotal comments on the questionnaire, it was determined by viewers in the Vancouver audience that the reason that the one child did not chant the pattern was due to shyness rather than her ability to actually perform the task.

The instruction transmitted from Australia was less successful. Children in the Vancouver studio were unable to perform the directional melodic task for two reasons. First, they were unfamiliar with the concepts of high and low, hence, the task was developmentally advanced. Second, they were unable to interpret the motions of the music specialist in Sydney because of the slow video resolution and because of the distant framing of the camera shot. Only when the Australian teacher invited the children in her studio to demonstrate the activity did the Vancouver children accurately perform the task.

Interpretations of the data

The studio setups in Vancouver and Sydney were quite different in their level of technological sophistication. The Vancouver studio had a centrally placed 40" monitor to view the incoming reception, a 20" monitor to view the transmission (placed to the left of the main monitor) and a 20" monitor to preview the transmission. All three monitors are in-wall units. Two video cameras were used to cover broadcast angles. The educational technologist used a touch screen Crestron Command Center to coordinate the audio and visual aspects of the session. A drawing of a yellow smiling face was placed next to the reception monitor to help focus the Vancouver children on the incoming transmission during their instructional sequence. The Sydney space had a 20" reception monitor for the Vancouver image and a 20" monitor for the Armidale video image. Both monitors were on movable carts. Essentially, whereas the Vancouver site is an enclosed studio, the Sydney site is an open, convertible space. The contrasting configurations and tools drastically affected rater perceptions. For example, responses on questions 3, 4, 5, and 6 (see Appendix A), where the video and audio signals were considered poor and severe by Rater 4 (Sydney), were considered average to excellent by Raters 1, 2, and 3 (Vancouver).

Essentially, the Australian rater (R4) believed that "the problems were not with actual teaching but with logistical problems at this (Sydney) end; volume was too low; linking three locations on two screens caused confusion—who should talk? who should we watch?; some of the words were indistinct; sound in and out intermittently." In contrast, Rater 1 (Vancouver) commented that "the first beat of the children's echo-chant is cut off, however, there seems to be no problem in the communication and interaction between the children and the teacher; the use of video conferencing technology for interaction between teachers and students in other countries is meaningful and has much potential."

An unsolicited, in-studio Vancouver observer who is versed in the technology wrote that "the quality of picture was more than acceptable for an across the Pacific transmission, with some limited blurring of the image when there was rapid movement being a minor problem rather than a major distraction." He felt that "the two teaching staff involved use the video conferencing medium with poise and assurance." Finally, he believed that the experiment "was able to demonstrate that the technology exists for successful real time two way video and audio communication across the Pacific."

Conclusions

Even though video resolution quality is not optimal with two line transmissions, it is a cost effective communications technique and distance education tool. Because of its cost effectiveness, additional research should be conducted to explore its use in fully compatible ISDN (p X 64 to p X 64; 15fps video resolution) contexts, which are now available in North America.

Arguably, even in its most sophisticated forms, children's television programming is essentially non-interactive. Video conferencing enables educators to explore the interactive aspects of audio (i.e., music) in an audio-video context. To that end, researchers are encouraged explore the efficacy of this potent technology with a variety of teaching techniques and in a variety of music and non-music contexts.

References

Burtenshaw, L. (1992, July). *Experiments in video conferencing (in Australia)*. Invited paper at the National Conference of the Australian Music Education Lectures' Association, Sydney, Australia.

Burtenshaw, L. (1993a, April). *Perspectives of distance learning in Australia: The potential role of video conferencing in music education*. Invited paper at the Suncoast Music Education Forum, University of South Florida, Tampa, FL.

Burtenshaw, L. (1993b, January). *Video conferencing--its implications for music education*. Invited paper at the Colorado Music Educators' Conference, Colorado Springs, CO.

Gouzouasis, P. (1993, April). *Music the medium: What's the message?* Refereed paper presented at the Canadian Music Educators' Conference, Wilfred Laurier University, Waterloo, ON.

MacKinnon, A. & Scarr, M. (1992, April). *The "Cariboo Connection."* A paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA.

McLuhan, M. (1964). *Understanding media: The extensions of man*. New York: McGraw-Hill.

McLuhan, M. & McLuhan, E. (1988). *Laws of media: The new science*. Toronto: University of Toronto Press.

Appendix A

The following are a sample of the rating scale items used in the evaluation of the experiment.

1) The live visual signal was ______

1	2	3	4	5
poor		average		excellent

4) The teacher's non-verbal directions (gestures to the children inviting them to chant) seemed to be clearly interpreted by the children

1	2	3	4	5
poor		average		excellent

5) The delay on the visual signal was ______

1	2	3	4	5
severe		average		none at all

6) The delay on the audio signal was ______

1	2	3	4	5
severe		average		none at all

9) Generally, the effect of the interaction of video/audio signal delay on the teacher's presentation and communication of information with the children was the most dramatic effect on the sessions that involved children.

1	2	3	4	5
completely disagree				strongly agree

10) The teacher's verbal directions were clearly presented.

1	2	3	4	5
not at all		fairly well		consistently

12) The teacher's verbal directions seemed to be clearly interpreted by the children.

1	2	3	4	5
not at all		fairly well		consistently

13) The teacher's non-verbal directions (gestures to the children inviting them to chant) seemed to be clearly interpreted by the children.

1	2	3	4	5
poor		average		excellent

14) The delay in the video broadcast seemed to be the cause of a problem in the children's echoing of the rhythm patterns.

1	2	3	4	5
on all patterns	on most patterns	only on some patterns	on very few patterns	not for any patterns

If possible, list the patterns which caused problems (refer to rhythm pattern list and write the number of the pattern). ______________________________

15) The delay in the video broadcast seemed to be the cause a problem in an individual child's echoing of the rhythm patterns.

1	2	3	4	5
on all patterns	on most patterns	only on some patterns	on very few patterns	not for any patterns

If possible, list the patterns for which there were problems (refer to pattern list and write the number of the pattern) for each child. ______________________________

16) The delay in the video broadcast seemed to be the cause of a problem in the teacher's sequencing of the rhythm patterns.

1	2	3	4	5
on all patterns	on most patterns	only on some patterns	on very few patterns	not for any patterns

If possible, list the patterns for which there were problems (refer to pattern list and write the number of the pattern) for each child. ______________________________

An Authoring System for Creating Computer-Based Role-Performance Trainers

DAVID GURALNICK and ALEX KASS

The Institute for the Learning Sciences
Northwestern University
Evanston, Illinois 60201

Abstract: In this paper we will describe a multimedia authoring system called MOPed-II[1]. Like other authoring systems, MOPed-II greatly reduces the time and expense of producing end-user applications by eliminating much of the programming effort they require. However, MOPed-II reflects an approach to authoring tools for educational multimedia which is rather different from most tools available on the commercial market. In this paper we argue in favor of this approach, which we call theory-rich authoring tools, and describe the success we have had with MOPed-II.

1. Introduction

The most common approach to authoring tools for multimedia applications is the general-purpose construction set, as exemplified by such commercial software programs as HyperCard, SuperCard, ToolBook, and Authorware. These tools are useful for building interface prototypes and simple applications. When used to build educational and training software, however, such tools leave it to individual tool users (that is, software developers) to structure the student's educational experience. The idea of using general-purpose (or theory-neutral) tools to build complex learning environments is predicated on the belief that tool users will be able to produce high-quality educational software if only they can be relieved of the programming burden. We believe that the task of designing as well as programming pedagogically effective software is a difficult challenge. Production of high-quality educational software requires more than technical achievement; it requires insight into what makes for an effective learning experience. Therefore, authoring tools for educational software should help developers design and build effective learning environments. We claim that theory-rich authoring tools which guide a developer in building a learning environment will help tool users build pedagogically sound software quickly and easily. In this paper we justify this claim and describe MOPed-II, a theory-rich authoring tool.

2. Why theory-rich authoring systems are needed

2.1 The importance of authoring tools

Computers have the potential to make great contributions to education and training, but that potential is difficult to realize. At its best, education and training software can, among other things, allow students to work on projects that would be infeasible without the help of computer simulation; provide individual feedback to the student at just the point when it is needed; and make a broad array of expertise and reference material easily accessible. Little such software is actually in use in the field, largely because of the size of the required software-development effort; an enormous amount of software will be required in order for the computer to make a noticeable impact, and high-quality software is expensive to produce. Thus there is an imposing economic barrier blocking the path from the laboratory to the classroom. Innovators may produce a few great pieces of software, but turning a few innovations into enough quality hours of instruction to account for a significant fraction of a curriculum remains a challenge. The problem is made particularly acute by the fact that high-quality educational software often requires technically challenging components such as complex interfaces, multimedia, and simulations.

---

[1]MOPed-II is based on an earlier authoring system called MOPed, developed by Enio Ohmaye (1992).

2.2 Theory-rich, not theory-neutral tools

We believe that a crucial step toward realizing the advantages that computer-based training can provide is to increase the number of people that can produce high-quality training software, via the development of authoring systems. But what form should those tools take? The tools currently available come in a variety of forms. Commercial drill-and-practice software tools can make it easy to construct a very narrow range of rigidly-structured learning environments, but those applications tend to be structured according to what is easy to implement on a computer rather than what makes for effective learning. Graphical user-interface tools are increasingly common; they provide the software developer with capabilities such as simple ways to put in graphics and buttons. But GUI tools solve only a portion of the software-building problem, and cause some other problems themselves. Johnson et al. (1993) describe several such problems, including that GUI tools require too much specialized programming skill and often enforce arbitrary forms of consistency (e.g., from Johnson et al., "everything is controlled via menus.") At their best, general-purpose tools are experts at managing computer objects. Educational software developers need tools which help manage pedagogy as well.

The goal of authoring tools for educational software should be to facilitate a clear conceptual mapping from the pedagogical objectives that an author wishes to achieve to an appropriate computer-based learning environment. General-purpose tools, particularly GUI tools, force software developers to attempt to map the conceptual components of their well-defined task to the physical components of a computer interface. This effectively takes the target audience of authoring tools, experts in teaching and in specific content areas, and forces them to act as novice computer programmers. The way to allow non-programmers to build good educational software is to give them tools that let them build software out of constructs they're familiar with, not to make them into pseudo-programmers. Authoring tools for educational and training software should take the form of special-purpose tools geared toward building specific classes of learning environments.

3. What types of theory-rich tools should we build?

3.1 Classifying educational software

At the Institute for the Learning Sciences, we have built a wide range of different types of educational software programs (brief descriptions of some of the Institute's software can be found in [Schank, 1991]), including systems for children (K-12) as well as for professional and military training. The tasks students must perform in our software are also quite varied. For example, some programs allow a student to create something new (e.g., CreAnimate, a biology tutor which lets students create their own animal–see Edelson, 1992), others let a student explore some subject area (e.g., Road Trip, a U.S. geography program which lets students simulate traveling across the country to reach destinations they've chosen–see Kass & Guralnick, 1991; Kass & McGee, 1992); still others have a more-specific task for the student, such as learning a foreign language (e.g., Dustin, a foreign-language tutor aimed at consultants who come to the United States for a training course–see Ohmaye, 1992).

Through our experience in building educational software, several classes of tasks have emerged as common–e.g., exploration tasks, artifact-construction tasks and "role-performance" tasks (defined below). In addition, pieces of software which teach different tasks within a particular class often share similarities in structure. We claim that the best way to construct authoring tools that encourage pedagogically appropriate design is to give them knowledge about task structures. We can then build theory-rich authoring tools for each structural class of educational software that we can identify. The tools' built-in knowledge about task structures helps them provide task-building guidance for software developers. In the remainder of this paper we will define one class of tasks and describe a tool we have built to build learning environments for tasks of that class.

3.2 Role-performance trainers: An example of educational software structure

We define role-performance tasks as tasks in which there is a fairly regular routine, such as operating a piece of equipment. At each point in a role-performance task, there is a correct next step (or small set of acceptable next steps) that the student should take. For example, the real-life task of working as a fast-food restaurant cashier and ringing up customers' orders on a customized cash register is a role-performance task. For each food item in a customer's order, there's a certain key sequence (or possibly a few options) for that item. A good order-ringer knows which keys correspond to which food items and can ring up orders quickly and accurately. Creative problem-solving is not the core job skill; understanding the procedure is. Role-performance tasks can come in large and more complex forms as well. The task of fielding customer complaint phone calls for a large water utility also has a regular routine, though it's harder than ringing up food orders. A customer service representative needs to learn which questions to ask a customer in order to diagnose the cause of the customer's water problem and recommend treatment. A good service representative understands an accepted set of common water problems, causes, and questions to ask a customer. So role-performance tasks involve learning a procedure of some sort, though it may have variations and options, and may be big.

Unfortunately, the procedural nature of role-performance tasks often causes people to try to teach them by rote. For example, one could try to teach cash-register skills by asking a student to memorize sequences of register buttons. Brown, Collins and Duguid's work on situated cognition (1989) showed that the activity during which knowledge is developed is an integral part of what is learned. Rote memorization separates knowledge from the real-life activity it is intended for. Instead, role-performance tasks are best taught by having the student practice the task and learn any required principles in context. Good role-performance trainers (RPTs) enable students to learn the task they are being trained to perform in an environment like the one in which they will actually perform it. Further, good RPTs guide students to understand how to perform the task.

3.3 Components of role-performance trainers

Building an authoring tool for a task class requires an analysis of what components the trainers for that class should have. Role-performance training breaks down into two primary components: practice and guidance. A good role-performance trainer includes a realistic simulation of the actual task for a student to practice on, along with tutoring to help the student understand his successes and (especially) his failures. The class of role-performance tasks includes two subcategories: theory-based role-performance tasks and convention-based role-performance tasks. All role-performance tasks represent procedures. In theory-based role-performance tasks, the correct steps in the task have some non-arbitrary reason for their correctness. A student who understands the underlying theory will be more successful at performing the task. For example, a water utility customer service representative who understands the causal model of how water reaches a customer's home is well-equipped to diagnose customers' water problems. In convention-based tasks, there is no deep theory behind how to perform the task. For example, a student need not go through a complex reasoning process in order to understand the positioning of buttons on a fast-food cash register. So a theory-based role-performance trainer might benefit from some components that a convention-based training system wouldn't need.

Role-performance trainers typically include the following components:

- **Simulation:** Since role-performance trainers allow students to learn by doing–that is, by practicing the task–a good RPT needs a realistic practice environment.
- **Failure Identification:** A role-performance trainer must guide the student in performing the task. When a student makes a mistake, the RPT needs to be able to explain the nature of the failure to the student as well as teach the student how to recover from the failure.
- **Scaffolding:** A role-performance training system must provide support, or *scaffolding* (Collins, Brown, & Newman, 1989), for a student who is performing a task. Scaffolding in a training system essentially means that the system performs or helps the student perform parts of the task that the student cannot.

Certain role-performance trainers also need the following additional components:

- **Many scenarios:** Most role-performance tasks have lots of variations. A guided practice environment for such a task is only useful if it provides lots of different scenarios for a student to practice on.
- **Socratic Tutoring:** Socratic dialogues help students understand the reasoning behind the steps of a procedure (Collins & Stevens, 1983). Training systems for theory-based role-performance tasks can especially benefit from a Socratic tutoring component.
- **Modeling:** It is often helpful for a student to learn how to perform a task in part by watching an expert do it (Collins, Brown, & Newman, 1989). This method is useful mainly for social types of tasks, such as speaking a foreign language (e.g., Ohmaye's [1992] foreign-language trainer, Dustin).

4. MOPed-II: An authoring system for role-performance trainers

MOPed-II is a theory-rich graphical authoring tool for building role-performance training systems. A software developer constructs a MOPed-II application out of conceptual components which represent tasks and subtasks that a student must perform. This is in contrast to general-purpose tools, whose program components are typically physical objects such as buttons or graphics. Rather than force software developers to try to convert content and teaching knowledge into a vocabulary of buttons and programming statements, MOPed-II lets them build the actual program using a vocabulary of task structures. MOPed-II's theory-rich structure guides non-programmers in building high-quality role-performance trainers quickly and easily.

4.1 A sample application built with MOPed-II

In order to understand how MOPed-II aids in build role-performance trainers, it is useful to consider an example. We have built a range of programs with MOPed-II, including a program which teaches foreign-born consultants how to speak English, and an application which teaches customer service representatives from a water utility how to diagnose and treat customers' water problems. Another program which we have built with MOPed-II teaches employees of a fast-food franchise to ring up customers' orders using the restaurant's special-purpose cash register. At the chain for whom this system was built, the job description of "cashier" includes only events which transpire after the customer's food has been collected and packaged. The cashier's job is essentially to ring up the sale on the cash register and collect money from the customer.

A large portion of the screen in this sample application is taken up with a simulation of the cash register. Both the keyboard and register screen look and act like the real-life register. The student can take any of three different types of actions at any point in a session with the program:

- press a key on the register
- say something to a simulated customer (e.g., "That will be $8.75, please.")
- ask the on-line tutor a question

One area of the screen displays a picture of the food order along with (sometimes) a text description of the items in the order. Another screen region holds two performance meters ("speed" and "accuracy"), which provide the student with feedback. Other sections provide buttons for the student to talk to the customer or ask the tutor a question.

At the beginning of each scenario the student sees a video, in which the "server" (the employee whose job is to package the customer's order) tells the student what the customer has ordered, just as the servers tell the cashiers this information in real life. Figure 1 shows a portion of a transcript from a typical student interaction with the cashier role-performance trainer:

Server (in video): The customer has ordered a half chicken and a small drink.
Student: Presses "Now What?" button
Tutor: You need to ring up the half chicken and ring up the drink.
Student: Presses "How Do I Do That?" button
Tutor: To ring up the half chicken, you need to press the "HALF CHICKEN" key. It is the key you see flashing on the register now. *[key flashes on screen]*
Student: Presses HALF CHICKEN key *[register screen shows half chicken rung up]*
Student: Presses $1.19 DRINK key *[register screen shows $1.19 drink rung up]*
Tutor *[intervenes]*: You don't need to ring up the large drink.
Student: Presses "Why?" button
Tutor: You don't need to ring up the large drink because it's not the right size.
Student: Presses "Now What?" button
Tutor: You need to void out the last item.
Student: Presses "How Do I Do That?" button
Tutor: To void out the last item, you need to press the VOID key twice. It is the key you see flashing on the register now. *[key flashes on screen]*

Figure 1. Sample interaction with the cashier-training application.

The cashier training system runs students through a series of scenarios, each scenario including a different customer order for the student to ring up.

4.2 The structure of MOPed-II applications

Computer-based role-performance trainers can be decomposed into three layers: the interface layer (objects on the screen that the student sees and manipulates), the task environment layer (the engine processing a student's actions, including the simulation component), and the tutoring layer (including hints, suggestions, explanations, demonstrations). MOPed-II supports building of the task environment and tutoring layers along with communication with an interface layer.

A scenario is a series of tasks that a student must perform. Each MOPed-II training application is composed of a number of scenarios. Typically, each scenario is a variation on the standard task of an application. For example, in the cashier training system, the overall task is to ring up customers' orders, and each scenario involves a different customer. Scenarios are composed of subtasks. The main subtasks for the cashier application are ringing up the items on the cash register, collecting money from the customer, ringing up the money correctly and giving the customer change (if any). The details of how and when a student is asked to perform each task in a scenario may vary depending on the student's performance; for example, a student who has not rung up the items correctly cannot move on to the money-collection task.

4.3 Creating scenarios

To create a scenario, an author using MOPed-II describes the structure of the task associated with the scenario by selecting a task-structure class from an on-screen palette. MOPed-II includes a set of predefined general task structures, called templates, which are common to role-performance tasks. One such structure is the "Unordered Steps" task template, which is used for tasks in which the student must perform certain steps, but not in any particular sequence (e.g., ring up the half chicken and ring up the drink in the example above). Each template provides a simple English-language form which lets the software developer fill out such information as the names of the steps in the task, English descriptions of the steps (for use by the tutor), and reasons why certain steps are correct or incorrect within the context of the particular task. Filling out this form yields a graphical task structure.

Each graphical task structure is composed of a sequence of actions. These actions come in four varieties:

- **output processing**: the system initiates some action upon their execution, such as show a video
- **input processing:** wait for input from the student before continuing processing
- **flow-of-control:** e.g., begin or end a task, branch based on some criteria
- **subtask:** move into another task or subtask

A developer need not utilize a template-generated task structure as-is. It is common for a developer to specialize the general predefined task structure for his own needs–e.g., providing special interface functionality. Figure 2 shows a typical task structure from the cashier training course described above. This structure represents the task from the example above in which the student's goal is to ring up the half chicken and ring up the drink. When a student has achieved that composite goal, the program goes on to the next task. If the student takes a wrong step, tutoring is provided (via a linked program called the Teaching Executive, written by Kemi Jona–see [Jona & Kass, 1993]), and the program enters the appropriate task for recovering from the error (e.g., the "void last item" task).

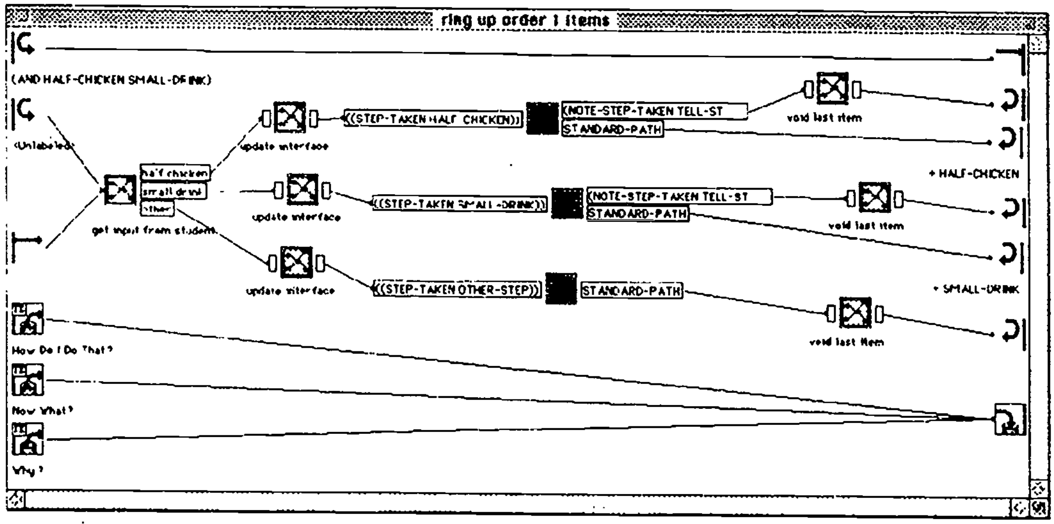


Figure 2. Cashier training course task structure.

So the general MOPed-II task-building process takes the following form: for each task, the developer chooses a general task structure template, fills out its English form, and (perhaps) specializes the task structure a bit more. Occasionally, MOPed-II applications might require a task structure which doesn't have a template. But building tasks from scratch is fairly simple, particularly when template-generated structures provide examples.

4.4 Generating tutoring

As mentioned above, tutoring in MOPed-II is controlled by a linked program called the Teaching Executive. The Teaching Executive provides an architecture for providing tutoring. Tutoring comes in two major forms: remediation (which requires failure identification)–telling a student he made a mistake and helping him understand why (possibly including Socratic tutoring); and support (scaffolding)–giving a student hints about what to do next, or demonstrating a step (possibly including modeling). Remediation and support can range from simple to complex, depending on the particulars of the task and steps the student needs help with. For example, a student who can't find the "Small Drink" key on a cash register simply needs to be shown where it is located, while a student who misdiagnoses a customer's water problem could benefit from a lengthier Socratic dialogue about the basis for his mistaken hypothesis. Most of the work needed to generate simple tutoring in a MOPed-II application is included in the task-building process. MOPed-II tasks include special structures which communicate with the Teaching Executive. The Executive itself houses modules which are called in a particular tutoring situation–e.g., the "Now What?" module, which is called when a student presses the "Now What?" button. More-complex tutoring forms such as Socratic dialogues are handled by separate, special-purpose modules which are invoked by the Teaching Executive at the appropriate times. As with task structures, a set of general Teaching Executive modules is provided. The application-builder chooses the modules he needs and fills in a small amount of application-specific information.

5. Conclusion

In order for computers to have a significant impact on education and training, people need to produce a large quantity of high-quality software. One way to generate lots of software is to expand the group of people who can build it. We can do this by building tools to allow people other than computer programmers to build software. In particular, we need tools for content experts and teachers to use. General-purpose authoring tools such as GUI tools force content experts to map their knowledge of the task to a set of computer interface objects–this turns out to be almost difficult as programming.

Instead, we need a set of theory-rich tools which are built especially for educational and training software, and guide software developers in building pedagogically sound learning environments. We have built MOPed-II, a theory-rich authoring system for role-performance trainers. We believe that theory-rich authoring tools are the best way to help non-programmers build high-quality educational and training software.

6. References

Brown, J.S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. Educational Researcher, 18, 32-43.

Collins, A., Brown, J.S., & Newman, S.E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In Resnick, L.B. (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum Associates.

Collins, A. & Stevens, A. (1983). A cognitive theory of inquiry teaching. In Reigeluth, C.M. (Ed.), *Instructional-design theories and models: an overview of their current status*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Edelson, D. (1992) Learning From Stories: Indexing and Reminding in a Socratic Case-Based Teaching System for Elementary School Biology. T.R. 43, Institute for the Learning Sciences, Northwestern University.

Johnson, J.A., Nardi, B.A., Zarmer, C.L., & Miller, J.R. (1993). ACE: Building Interactive Graphical Applications. *Communications of the ACM, 36*, 41-55.

Jona, M., & Kass, A. (1993). The Teaching Executive: Facilitating Development of Educational Software through the Reuse of Teaching Knowledge." In Estes, N. & Thomas, M. (Eds.), *Proceedings of the Tenth International Conference on Technology and Education* (pp. 640-642). Cambridge, MA: Massachusetts Institute of Technology.

Kass, A., & Guralnick, D. (1991). Environments for Incidental Learning: Taking Road Trips Instead of Memorizing State Capitals. In L. Birnbaum (Ed.), *Proceedings of the International Conference on the Learning Sciences* (pp. 258-264). Evanston, IL: Assoc. for the Advancement of Computing in Education.

Kass, A., & McGee, S. (1992). The Road Trip Project: Learning Geography through Simulated Travel. Technical Report 42, Institute for the Learning Sciences, Northwestern University.

Ohmaye, E. (1992). Simulation-Based Language Learning: An Architecture and a Multimedia Authoring Tool. Technical Report 30, Institute for the Learning Sciences, Northwestern University.

Schank, R.C. (1991). Case-Based Teaching: Four Experiences in Educational Software Design. Technical Report 7, Institute for the Learning Sciences, Northwestern University.

7. Acknowledgments

We wish to acknowledge the members of the MOPed-II development team and users' group, including Laura Allender, Mark Chung, Scott Dooley, Jonathan Hickey, Jeff Lind, Marek Lugowski, Frank Luksa, Inna Mostovoy, Suzanne Pink and Wayne Schneider. MOPed-II developed out of an earlier authoring tool called MOPed, designed and built by Enio Ohmaye and Mark Chung. Kemi Jona designed and built the Teaching Executive and also contributed greatly to MOPed-II and several applications built with it. Ann Kolb, Jarrett Knyal, and Josh Tsui contributed excellent graphics work. Chip Cleary provided valuable comments on earlier drafts of this paper. We would also like to thank Roger Schank for his leadership.

This research was supported in part by the Advanced Research Projects Agency, monitored by the Office of Naval Research under grant no. N00014-91-J-4092. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting, part of The Arthur Andersen Worldwide Organization. The Institute receives additional support from Ameritech and North West Water Plc, Institute Partners.

Interactive Video and Sign Language for Improving Literacy Skills of Deaf Students

E. HANSEN, J. MOUNTY, A. BAIRD
Research Division
Educational Testing Service, Rosedale Rd., Princeton, NJ 08541, USA

Abstract: Students who are deaf often have great difficulty in accessing written English. A "bilingual" approach to education may help deaf students improve their literacy skills. This paper explains work in progress regarding two projects supported by the United States Department of Education, which explore the potential of interactive video technology and sign language for improving reading comprehension and test-taking skills of deaf junior high and high school students. These projects feature: student access to one or more signed versions of English to help them understand any portion of the target text; a sign language dictionary of difficult words or phrases; instruction, questions, and corrective feedback provided in sign language as well as text. Use of the multimedia technology and sign language appear highly motivational to students. A number of factors seem to influence students' capacity to benefit from sign language helps.

PROBLEM

Despite dedication on the part of educators during the more than 150 years since educational programs for the deaf were established in this country, most deaf individuals are far behind their hearing peers in the acquisition of literacy and knowledge (COED, 1988; Moores, 1987; Moores & Sweet, 1990; Marschark, 1993). Research shows that average 18- to 19-year-old, severely to profoundly deaf students are reading no better than the average 9- to 10-year-old hearing students (Paul & Quigley, 1990). This fact is often a great surprise to those not acquainted with deafness; it is commonly assumed that individuals who are deaf should have no special difficulty in understanding written English, unless they are also visually impaired. Nevertheless, this lag is a harsh reality and presents a great barrier to academic and employment opportunities for deaf individuals.

Researchers have considered various explanations for the cause and nature of the difficulties that deaf readers encounter with English text. It has been diversely suggested that the focus of the problem lies in semantics or syntax, or in sentential level comprehension, or in discourse level interpretations of text (Moores, 1987; Paul & Quigley, 1990). Findings remain inconclusive at best, although it is safe to say that the problems and solutions are complex. At the heart of the matter is the reality that most deaf children begin school insufficiently grounded in rich early language and learning experiences (Martin, 1985) and are further set back during the school years by insufficient access to the language and culture of the educational system (Vernon & Andrews, 1990).

Thus, in summary, English most often is not fully accessible to deaf students as a first and/or primary language. Consequently, deaf students generally experience extreme challenges in mastering English literacy. These inadequate literacy skills contribute to poor performance on standardized tests and often limit opportunities for postsecondary education and for employment.

Possible Solutions

Fortunately, there may be solutions or partial solutions on the horizon. Among them are: (1) educational tools that utilize deaf students' competencies in sign language that may help build English skills; (2) advances in interactive video and computer technologies that provide new possibilities for teaching and learning; (3) bilingual-bicultural approaches that may allow the development of new techniques to enhance language-related thinking.

For those individuals who sign, it seems that access to sign via interactive videodisc technology might facilitate understanding of reading comprehension tasks, such as those which appear on standardized tests. Despite research that has examined the issue of sign comprehension by deaf persons of various backgrounds (Caccamise & Gustason, 1979; Caccamise, Hatfield & Brewer, 1979; Quigley & Paul, 1984; Johnson, Liddell & Erting, 1989; Bochner & Albertini, 1988), the data remain inconclusive. An important issue warranting further investigation concerns which kinds of signing to use in specific learning contexts.

PURPOSE

Under support from the U.S. Department of Education, Educational Testing Service is carrying out two projects that explore the use of interactive video technology and sign language as "levers" for improving the learning, performance, and diagnostic assessment of deaf individuals. It is hoped that these projects will help define, as well as improve, the kinds of accommodations that can and should be provided to deaf individuals in academic and professional testing and certification situations. The major research goals of the projects are as follows:

1. Explore the use of sign language and interactive video for improving English literacy and test-taking skills.
2. Observe how student characteristics, environmental influences (e.g., school language policy), and content variables (e.g., type of passage) may relate to students' preference for ASL or English-based sign (a version that adheres more closely to English word order).
3. Observe how each signed version assists comprehension.
4. Explore bilingual-bicultural approaches for enhancing language-related reasoning and the literacy that builds upon it.
5. Lay the groundwork for improved tools for instruction, assessment, and performance support.

Background

Prior to the two current projects, some of the concepts of the project were tried in a demonstration system in 1991. This proof-of-concept system uses sign language and interactive video. The main display of the application shows three "windows" -- one for scrollable passage text, one for the questions, and another for either of two sign language versions to be displayed in motion video. A variety of controls are available, mostly at the bottom of the screen, and are activated by mouse. In order to see a signed version of some English text, the student selects a portion of English text on the computer using the mouse and then selects either of two signed versions -- American Sign Language (ASL) or English-based sign. The system then displays a video of a person signing the material.

METHOD

The two new projects supported by the U.S. Department of Education build on the 1991 Demonstration System. These projects are aimed at deaf readers who are fairly proficient.

Hardware and Software

The system for both projects is based on a Macintosh platform. The system uses a Macintosh II-family computer, with a video graphics overlay card that allows full-motion color video from a Pioneer laserdisc player. The video is displayed in a resizable window on the computer screen. We are using the RasterOps 24STV video graphics overlay board and a Pioneer LDV-4200 laserdisc player. This and several other Pioneer player models allow one to jump to and begin playback of any section of the video in less than one second, making it easy to display any signed passage on demand. We are also using Quicktime movies for selected portions of the system and may store that compressed video on CD-ROM. The application software is being developed using the SuperCard (Allegiant Corporation) authoring software.

The Projects

Project 1: Test Preparation

Project 1, "Sign Language and Videodisc for Test Preparation Materials for Deaf Secondary School Students" (Grant No. H133C20006-92), focuses on the issue of test preparation. The audience for this one-year project is deaf high school students. The project is intended to demonstrate the feasibility of using ASL and video technology to help students prepare for standardized tests, especially tests of reading comprehension. The system provides a total of about 30-45 minutes of interactive instruction plus an online sign language dictionary. Small-scale pilot testing was carried out in the fall of 1993.

Project 2: Tools and Techniques for Improving Literacy

Project 2, "Videodisc Technology and Sign Language for Improving Reading Skills of Deaf Middle School Students" (Grant No. H180G20021), focuses on a variety of issues related to the improvement of literacy skills in deaf students. This three-year project focuses on middle school students. This project has two major phases.

1. Phase 1. Phase 1, to be completed in the mid-1994, entails an experiment to learn more about (a) factors that affect the impact of sign language helps (i.e., accessible sign language versions) on student performance and (b) indications of student preference for the particular signed versions -- ASL and an English-based sign. Each subject performs two tasks. For each task the subject reads a reading comprehension passage and then, using sign language, retells the passage to a deaf signer. For one of the two tasks, the subject then re-reads the passage as necessary and answers four or five multiple choice questions without any access to sign language helps. For the other task, the subject rereads and answers the multiple-choice questions with full access to both ASL and English-based sign versions of both the passage and the questions. Thus, each student sees two reading sets (passages plus questions), one with no access to sign language and the other with access to both signed versions, thereby allowing us to estimate the performance benefit that might result from such access. In order to balance out possible order effects such as fatigue and practice, the experimental design is balanced for both (1) order of conditions and (b) order of sets. It is expected that 24-48 subjects will be involved in the experiment. This experiment is in progress currently and is expected to be completed by early summer 1994.
2. Phase 2. Phase 2 of Project 2 explores techniques and approaches for bilingual and bicultural education. This phase will build on the current systems but will also include: (a) student video-production projects utilizing acting, signing, and writing; (b) computer exercises in reading; and (c) an enhanced online dictionary.

The Challenge of Translation/Production

One of the most significant challenges of the projects is the process of translation/production. The terms "translation" and "production" are joined because we find that much of the translation and production need to occur together in the same sessions. The process of translating English text to ASL and English-based sign usually requires a team of several people, who represent expertise in the following areas:

- Linguistic knowledge of both ASL and English.
- Native or near-native ASL competence
- Native or near-native English competence
- Acting skill and experience
- Video production with deaf actors
- Management and coordination

Although the actor and director can "draft" the translations prior to video recording, the translation cannot be readily or fully transcribed -- rather the "practice shots" or "drafts" must be videotaped and critiqued. What seems to work well is for the director and actor to review the English source material independently, and then come together to negotiate the subtleties of an acceptable signed rendition of the material. We find that the camera person must be ready to capture the translation. Sometimes a "first draft" or "practice shot" turns out to be "perfect," so it is critical to be ready to record.

PRELIMINARY RESULTS

Although we are currently in the process of gathering data, we have made some tentative observations.

- The Motivational Impact of Interactive Video. Students expressed enthusiasm about working with computers, especially interactive video.
- The Motivational Impact of Sign Language. Students appreciate seeing examples of excellent signing. Most are children of hearing parents who do not have excellent signing skills. Most of their teachers are hearing and have only moderately good skills. To see educational content signed by deaf signers with excellent signing skills is highly motivating.
- Factors That May Affect the Benefit of Sign Language Helps. Preliminary impressions of student use of the technology suggest that there are a variety of skills and factors that may influence the amount of benefit to be initially derived from sign language helps. Some of these factors are as follows:

  1. Sign Language Comprehension Skills of Students. Students appear to vary widely in their receptive skills in ASL and English-based sign; these skills would have a major impact on their ability to benefit from the sign language helps.
  2. Level of Prior Language-Related Reasoning Experience. While the incidence of specific language disabilities may not be higher in the deaf population than in an otherwise comparable group of hearing students, deaf students -- most of whom have hearing parents -- may have had less experience in language-related reasoning and hence may perform less well on language-related tasks, even when sign language helps are available. (For this reason, the project plans to include assessment of students' sign communication proficiency and comprehension of passages presented in sign language.)
  3. Prior Knowledge of the Topics. Students who are unfamiliar with the content in the reading comprehension passages may be unfamiliar not only with the English words and phrases but also their sign language equivalents or analogs.
  4. Knowledge of the Operation of the Computer System. The usefulness of the sign language helps is obviously dependent on students' understanding of the operation of the computer system. Most students seem to have little if any problem with the operation of the system. However, early in the field testing it became apparent that at least one student did not make use of sign language versions of questions because she did not realize that help was available for questions just as it was for passages (although an online tutorial attempted to make that point clear). Subsequently, the deaf research team member who administered the field test stayed with the student for the start of this task to be sure that it was clear that such help was available.
  5. Beliefs About the Usefulness or Appropriateness of Sign Language. The usefulness of sign language helps can be influenced by the student's beliefs about the usefulness or

appropriateness of sign language for particular purposes. For example, one student simply did not use the sign language helps at all because she did not believe that it was appropriate to use sign language in the context of a reading comprehension task. Furthermore, some students may have persistent preconceptions about the relative value of a particular version of sign language over another; this was one reason that we chose not to designate the two versions by the terms "ASL" and "English-based sign." We did not wish to have the results influenced by students' prior beliefs about various forms of signed communication.

CONCLUSIONS

The current projects appear to confirm the highly motivational impact of multimedia and of sign language materials for deaf students. The current stage of the research effort does not permit solid conclusions about other benefits of sign language helps on instruction, performance support, and assessment. However, it appears that a number of challenging factors must be addressed in any system designed to leverage students' sign language ability in order to improve their English literary skills. Through continued research, we hope to learn how to address those challenges so that some variation on this approach can become a high-yield application of multimedia and other computer technologies.

REFERENCES

Bochner, J. H., & Albertini, J. A. (1988). Language varieties in the deaf population and their acquisition by children and adults. In M. Strong (Ed.) Language learning and deafness. New York: Cambridge University Press.

Caccamise, F., & Gustason, G. (Eds.) (1979). Manual/simultaneous communication instructional programs in the educational setting. Proceedings of a Conference, Kansas City, KS, Nov. 6-10, 1977. Washington, DC: Gallaudet College Press.

Caccamise, F., Hatfield, N., & Brewer, L. (1979). Communicating with hearing-impaired people. In Legal rights of hearing-impaired people. Washington, DC: Gallaudet College Press, National Center for Law and the Deaf.

The Commission on Education of the Deaf (COED). (1988). Towards Equality: Education of the Deaf. A report to the President and Congress of the United States.

Johnson, R. E., Liddell, S. K., & Erting, C. J. (1989). Unlocking the curriculum: Principles for achieving access in deaf education. Gallaudet Research Institute Working Paper 89-3. Washington, DC: Gallaudet University.

Marschark, M. (1993). Psychological development of deaf children. New York, NY: Oxford University Press.

Martin, D.S. (1985). Cognition, education and deafness. Washington, D.C.: Gallaudet College Press.

Moores, D. F. (1987). Educating the deaf: Psychology, principles, and practices (3rd ed.). Boston, MA: Houghton Mifflin.

Moores, D. F. & Sweet, C. (1990). Factors predictive of school achievement. In D. Moores & K. Meadow (Eds.), Educational and developmental aspects of deafness. Washington, D.C.: Gallaudet University Press.

Paul, P. V., & Quigley, S. P. (1990). Education and deafness. White Plains, NY: Longman.

Quigley, S. P., & Paul, P. V. (1984). Language and deafness. San Diego, CA: College Hill Press.

Vernon, M., & Andrews, J. F. (1990). The psychology of deafness. White Plains, NY: Longman, Addison-Wesley.

ACKNOWLEDGEMENTS

The work described has been supported by the Office of Special Education Programs (OSEP) of the United States Department of Education, grant numbers H133C20006-92 and H180G20021.

HyperCard Monitor System

JULIAN HARRIS, HERMANN MAURER
Hypermedia Unit
Computer Science Department
The University of Auckland, Private Bag, Auckland
Email: J_Harris@cs.aukuni.ac.nz, H_Maurer@cs.auckland.ac.nz

Abstract: An investigation into high level event monitoring within the scope of a well-known multimedia application called HyperCard is carried out. An attempt is made to gather and process user's actions at a high level, enabling accurate and quantitative statistical analysis to be performed. This type of information is either very difficult or impossible to obtain through other methods.

Introduction

In the computer world today, monitor systems are a reasonably recent development. For the purposes of this paper, we shall define a monitor to be a system which automatically monitors usage of some activity and gathers statistics based on what it has observed. Monitor systems are useful tools in many areas. For instance public computer-based kiosk systems have been used which gather information on what the user did while using the system (Maurer, Sammer & Schneider 1993). The system they employed was automatic but only involved audio and video recordings of what users did, thus making analysis a tedious manual process. Monitor systems can give the authors of a program or system that is being monitored very accurate statistics on *how their product is being used*. From the resulting information, they can assess what aspects of a product need the most attention.

One reason monitor systems are rare is the ratio between the scope of the monitor (what sort of information it collects) and the type of information. The more you limit the scope of a monitor system (eg. a system to monitor activity for a particular application) the easier it becomes to record useable information. However, the usefulness of such a specialised analysis tool is limited and in most cases infeasible. The most useful monitor system would be integrated with the Operating System (OS) itself, so that activity of every program running under the OS can be monitored. The main problem with this is that the only information the monitor system would obtain is information that is available to the system, which is usually very low level and of little general analytical value.

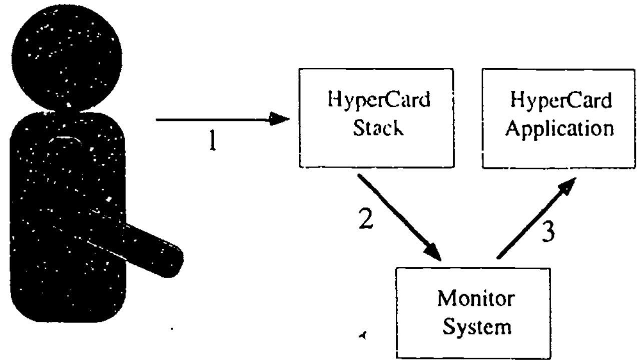
Currently the only way to find out how people are using a system is to have other people, cameras, or microphones observe the users, possibly having to record the data manually. This is problematic because of several reasons. One is that the mere fact that the observers are there will influence how the users behave, introducing unwanted biases. Another point is that this method is very costly, time consuming and accurate results can be very difficult to obtain. This is usually because of the distance between the data generator (the user) and the data gatherer (the observer). For example, a common approach is to ask the user at the end of the session to fill out a questionnaire or put down any comments they have about it. This approach will only yield general information, and will tend to be fairly inaccurate. A suitably designed computer monitor system will be able to retrieve the data *directly* from the source which will vastly improve the integrity of the data and will be able to catch much more useful information as well.

On the Macintosh Computer, there is a program called HyperCard which is a simple, easy to use multimedia system (Apple HyperCard, 1988). There is an abundance of programs (called stacks) created using HyperCard. The aim of this paper was to develop a simple HyperCard-based usage monitor which monitors Macintosh users as they run HyperCard stacks, and stores the resulting data away for later statistical analysis. In doing this, the scope of the monitor system is being limited to user activity that goes on within the HyperCard application while enabling us to easily catch high level information.

Overall Structure

A HyperCard stack is made up of objects arranged in a hierarchy. A stack owns a collection of backgrounds. Backgrounds own a collection of cards, and cards own a collection of fields and buttons.

When a user does anything in HyperCard, messages are generated which are sent to these objects through a message passing hierarchy (for instance, clicking somewhere generates a mouseup message). Each object has a script, within which can be a series of message handlers which are executed when the appropriate message is sent to that object. Message handlers consist of a series of English-like instructions (HyperTalk) like "Get the first field / if it is empty then answer 'That field is empty.'"



1 User Creates a HyperCard message (via mouse or keyboard or other input device)
2 Once the stack has handled it, it is passed on explicitly to the HyperCard Monitor
3 Whether or not the message is trapped by the Monitor, the event is then passed on to the HyperCard Application

Figure 1. How messages are caught by the HyperCard Monitor System

In this message handler, the message can be passed on further down the hierarchy for further processing. Handlers have to explicitly pass the message on if they want other objects further down the message hierarchy to get the message — it is considered good practise to pass the standard HyperCard messages on for other objects.

Our HyperCard monitor works by inserting itself in the HyperCard message-passing hierarchy (Figure 1) and sets up handlers for all the messages that are considered useful for data gathering. This requires that all messages the monitor handles are passed explicitly by the object that got the messages in the first place.

As many stacks do not actually pass these messages in all of their handlers, one thing that had to be done was create a utility which would modify a given stack to ensure that it would pass enough messages for our monitor to work since the monitor expects events to happen in a certain order (see Figure 2).

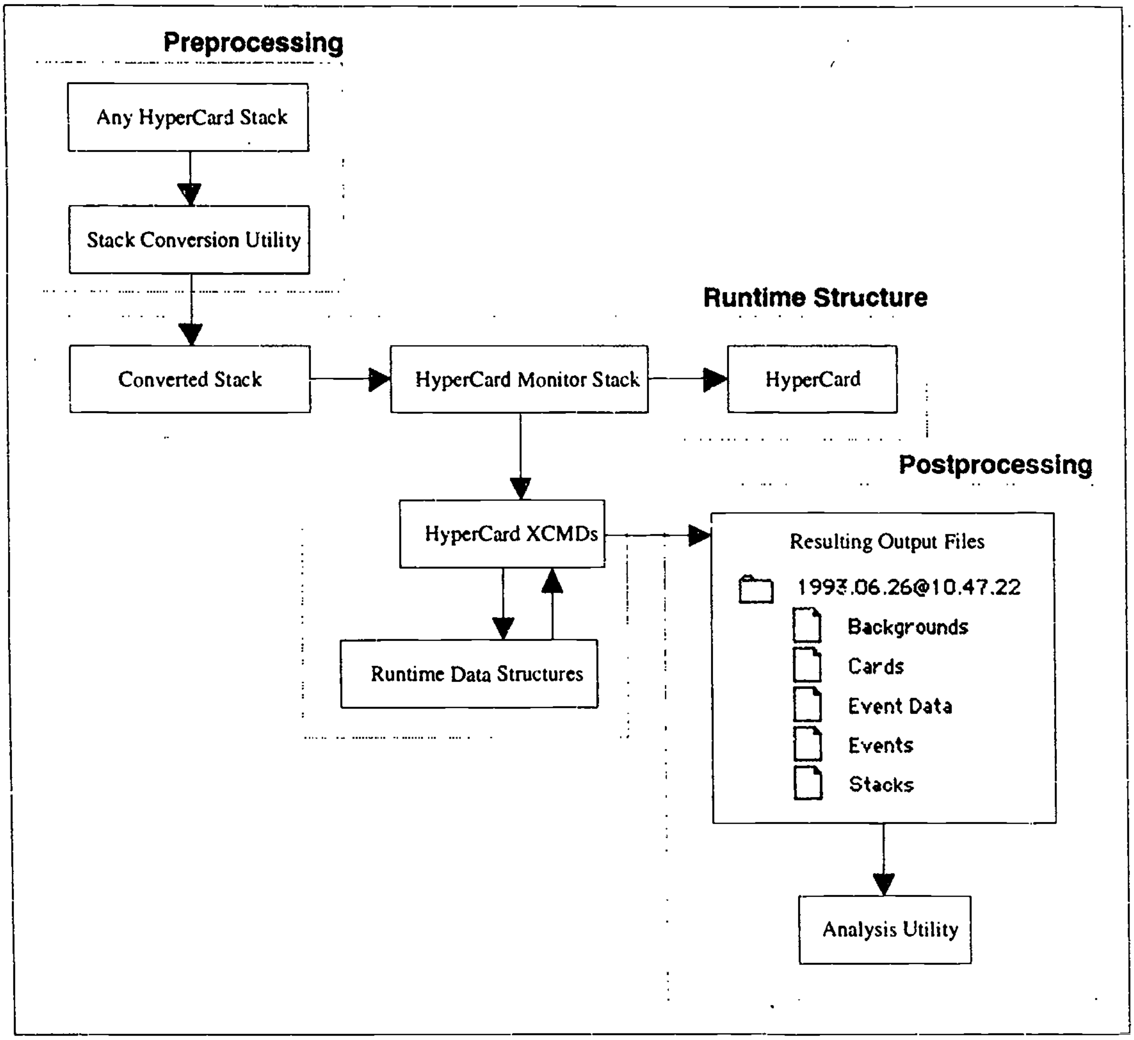


Figure 2. An overall view of the HyperCard Monitor System Structure

The Monitor System

There are many levels of statistical information that can be derived from the monitor data. One is analysis of a particular user for a particular work session. Another is head to head comparison of users over many sessions which involves more general statistical analysis. In systems like the proposed University Transaction, Information and Communication System (UTICS) (Schneider, 1993) which use fundamentally new interfaces between computers and humans, both head-to-head comparisons and overall statistical information is useful to assist in research when seeking to refine or remodel the interface.

Whatever data is analysed, there need to be clearly defined points which say when a user stops using the system and another user starts. This is a general problem and usually schemes like finishing a session after a predetermined period of inactivity are employed.

We defined the start of a session to be when the user starts up HyperCard (suitably modified so that it launches our monitor), when the user opens the monitor directly, opens a stack from the home stack, or finally when the user explicitly chooses to start a session using the monitor configuration. Most of the time it is usually one of the first two. Finishing a session is defined to occur when a user closes the monitor, quits HyperCard, goes back to the home stack, or explicitly chooses to finish the session using the monitor configuration.

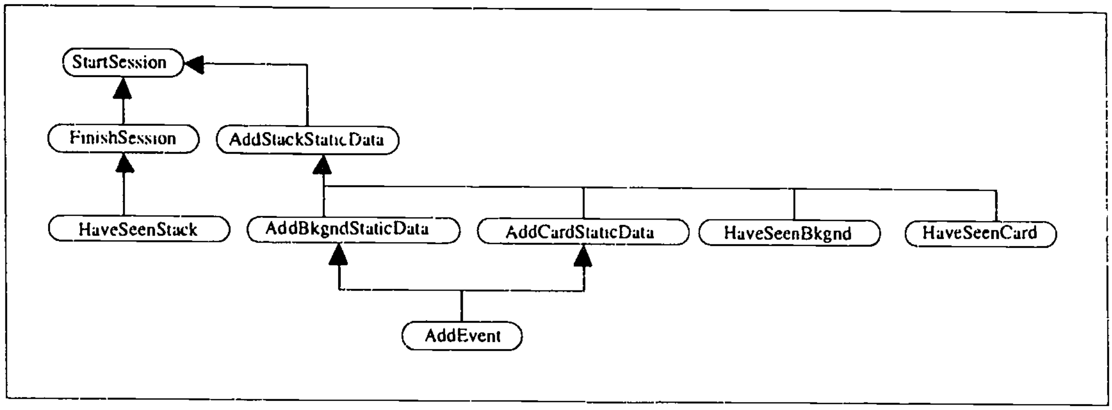


Figure 3. How the different XCMDs rely on each other

HyperCard can be extended through the user of external code modules called XCMDs (pronounced 'x-commands') (Coodman, 1990). The XCMDs are usually written in C or Pascal and are useful to implement things HyperCard is lacking. The monitor is internally divided up into two levels: a series of 11 XCMDs to do all the low-level grind work, and a HyperCard stack called "HyperCard Monitor" with associated HyperTalk handlers to capture the data and call the low-level XCMDs. Implementing most of the monitor in external XCMDs helps to keep it portable to other environments and improves the speed of operation considerably.

10 of the 11 XCMDs control access to the runtime and static data structures that are created during a session. The XCMDs are optimised for speed and most operations are independent on the size of the stack. Those that are not, employ caching mechanisms to improve performance.

One important issue is that the XCMDs are called in a clearly defined order. Figure 3 shows the structure of the first 10 XCMDs by name and a matrix showing which XCMDs have to be called first in order to have well-defined results. The 11th XCMD is the ConvertScript XCMD which makes a script fully compatible with the monitor. The monitor has a section which calls the ConvertScript XCMD to convert a specified stack.

Findings

Whether or not the method of recording user activity employed in this system is a useful approach to monitor systems in general is a good question. There were many key features which made the HyperCard environment a very good basis for such a system, and one was the ability to very easily install methods which would catch events as they happened. What was even better was that the events were at a level which were regarded as most useful when monitoring user activity. From this, it is easy to see that a way that a more general monitor system for an OS could be developed is by letting the OS communicate to and from programs in a similarly high level way. Apple has developed a system-level scripting system called AppleScript (Apple: AppleScript, 1993) which plugs in on top of Apple's System 7 OS. AppleScript provides an interface available to all applications which lets applications send high level messages to each other. For instance "move the selected object 3 pixels to the left and 14 down", "get me information on the person called 'Langley'", or "Get the contents of whatever the user selected".

Although the monitor system was implemented using a combination of HyperTalk scripts and XCMDs, many of the ideas and a lot of the code can be directly used in the more general implementation of a system-level monitor system, including concepts like the caching mechanisms, event data encoding, hierarchical data, and static data models. A system where records were kept of activity in each application, along with sub-records which kept track of windows in each application could be one approach. Another key feature an OS-based monitor system would have would be consistency, and clearly defined ways of performing activities. If there are clearly defined ways of doing things and they are done similarly in different situations, then the monitor system will be able to capitalise on this because it will be able to make assumptions. Assumed knowledge will help the system filter through irrelevant data and derive useful information from minimal data.

The idea of dynamic and static data could be applied as well — there are fixed attributes applications and to a lesser extent windows which can be stored once. In general, the content of application windows is more flexible than the contents of a card or background, and so perhaps another approach to storing the information in a window would need to be developed. Apple has developed a database of 'things that different types of applications can be told to do' which has thousands of entries, in a hierarchical structure. When people want to extend a message, a class-like approach can be taken so that the message is still identified as a member of the superclass, but is still identifiably different.

Overall, we consider the HyperCard Monitor System to be very effective and many of the general concepts used for the system could easily be extended to OS level if such an OS provided a facility for generating and interpreting high level events of this nature. Performance on a Macintosh Quadra 700 was very good and the resulting data output was very compact — tests revealed about 50k of data per user per $^1/_2$ hr which was several orders of magnitude better than initial prototypes.

One thing that was accepted when doing a monitor system in HyperCard is that there will be constraints and limitations imposed upon what, how and where the monitor can record data, and also to what level of detail and how quickly. There are several areas which could be investigated further to improve the performance and usefulness of the monitor system. Although HyperCard is inherently slow, the monitor's overhead is quite low due to the use of XCMDs and could be improved even more. Most of the speed improvements could come from more clearly defined states, storing less data, storing data in a more compact form, and reducing disk access through caching. A costly process is writing the static data for a card to disk. Often people browse through cards in a stack quickly and so a memory cache of recently used card data could be very helpful.

Another time-consuming process is verifying whether or not a stack has already been seen. Currently this is an O(n) comparison of strings which can in themselves be quite long. This is done every time any of the XCMDs are called. If we could find another way of identifying stacks (perhaps with some simple HyperTalk code keeping information attached to a stack) then performance would improve that way too. In-depth analysis of the monitor's output data could be used to ascertain what data turns out to be the most useful, thus discarding any relatively useless data and keeping the most relevant data more compactly.

A constraint enforced on the current version of the monitor system is that the monitor does not work well with stacks whose structure is altered. Specifically, cards, fields, or buttons added, deleted or sorted are unlikely to work because the runtime information is defined statically whenever a new stack is opened. To add support for addition and deletion of cards, fields, and buttons are non-trivial because of the implications of what a change in the ordering of the stack will do to the references in the data which has already been written to disk.

Some other miscellaneous and all together reasonable constraints are that the disk the monitor is on must be writeable and have enough disk space to accommodate the number of sessions intended to be run on it.

References

Apple Computer, Inc. (1988). The Macintosh HyperCard User's Guide,

Apple Computer, Inc. (1993). Inside Macintosh: AppleScript Language Guide,

Apple Computer, Inc. (1993). Inside Macintosh: Interapplication Communication,

Goodman, D. (1990). The Complete HyperCard XCMD Handbook for HyperCard 2.0 (3rd Edition).

Maurer, H., Sammer, P. & Schneider, A. (1994). Multimedia Systems for the General Public: Experiences at World Expositions and Lessons Learned, *Proceedings of 2nd Interactive Multimedia Symposium, Perth, Australia,* 333-441.

Maurer, H., Schneider, A. (1994). New Aspects of a Hypermedia University Representation, *Proceedings of ED-MEDIA '94,* Vancouver, Canada

A Hypermedia Information System For Aviation

KARIN M. HARTZELL
Galaxy Scientific Corporation
2310 Parklake Drive Suite 325, Atlanta, Georgia 30345
E-mail: mathcs.emory.edu!khartzel%galsci.UUCP

Abstract: The Hypermedia Information System (HIS) is being developed under the auspices of the Federal Aviation Administration (FAA) Office of Aviation Medicine's (AAM) Human Factors in Aviation Maintenance (HFAM) research program. The goal of the hypermedia project is to create new tools and methods for aviation-related information storage and retrieval. The HIS is currently undergoing transition from a proof-of-concept system to a fully functional system capable of supporting the instructional, maintenance, and safety inspection needs of the aviation industry. The next step is to increase its capabilities to support existing and emerging documentation standards.

Introduction

The aviation industry manages large quantities of documentation for many purposes, including training, research, maintenance, and safety inspection. These paper or microfiche documents include maintenance manuals, federal aviation regulations, and research reports. Timely and convenient access to these documents is important, but currently it can be quite cumbersome. For example, safety inspectors and aviation maintenance technicians must carry literally stacks of documents to the flightline when they inspect or repair an aircraft. Finding the desired information in these documents is not always easy and, therefore, not always accurate. Improvements in the way aviation personnel access this information will lead to more reliable and more cost-effective aircraft maintenance.

Toward this end, the Federal Aviation Administration (FAA) Office of Aviation Medicine (AAM) Human Factors in Aviation Maintenance (HFAM) research program is studying the challenges associated with creating, accessing, and maintaining digital documentation using a Hypermedia Information System (HIS). This paper discusses the current state of the HIS, including its interface features, system design and development, and its integration into multiple tutoring and job aiding systems.

The Hypermedia Information System Features

Over the past several years, Galaxy Scientific has been using hypermedia technology to address the challenges of improving aviation information access. After surveying commercially available products, Galaxy developed its own set of tools that could be distributed to the aviation community. Initial efforts concentrated on demonstrating the feasibility of a hypermedia system for aviation. Team members designed a digital library system and implemented rudimentary tools for information storage. The bulk of the implementation effort focused on the information retrieval tools and the hypermedia reader interface.

Federal Aviation Administration research reports were used as a testbed for creating the digital library. The proof-of-concept hypermedia viewer (FAA/AAM & GSC, 1994) proved to be a flexible and powerful way for researchers to view hypermedia documents. The HIS can be used alone as a tool to access information, as well as integrated with training and job aiding systems (Johnson and Norton, 1992). Both the viewer and the library were distributed on compact disc, read-only memory (CD-ROM) to the aviation maintenance community in early 1993. As with many proof-of-concept systems, this one was geared toward a specific application area. The viewer interface was tailored to the FAA research reports, making its broad-scale applicability limited. During the last year, research has continued, with an eye toward making the tools more generic and functional.

The HIS reader interface maintains a book paradigm and consists of two distinct components: a navigation component and a viewing component. The navigation component combines the familiarity of traditional book navigation, such as a table of contents, with the power of hypermedia searching. The viewing component allows the reader[1] to read, print, and manipulate the various media that make up the library.

Navigation

A traditional paper book provides several methods for navigating such as a table of contents, an index, and simple page turning. Likewise, the HIS supports a variety of access paths into and within a document. Some readers seek specific topics of interest and demand a powerful method for browsing a complex document. For them, the hierarchical Outline Viewer and query capabilities are useful. Other readers seek quick reference to standard information. For them, Hot Links and Bookmarks provide mechanisms for quickly going to frequently referenced places in a document.

The Bookshelf. The first component of the HIS the reader encounters is the Bookshelf, as shown in **Figure 1.** The Bookshelf is a graphical depiction of the various libraries available to the reader. The reader chooses a library by selecting an icon. The Bookshelf icons can be customized for a specific application.

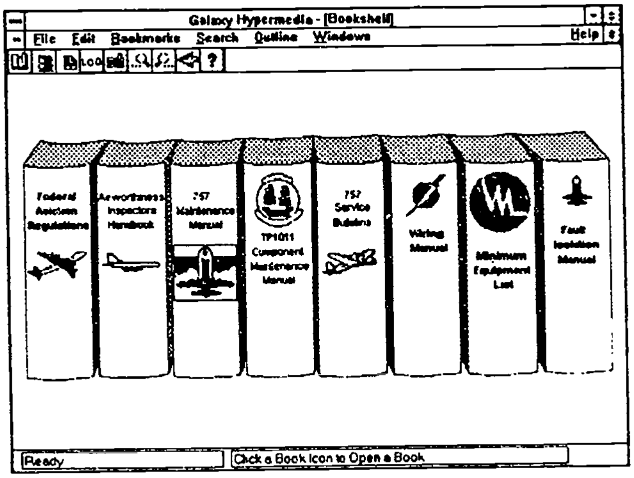


Figure 1 The HIS Bookshelf

The Outline Viewer. Once a library has been chosen, the Outline Viewer appears, displaying the complete outline for that collection of documents. The outline is similar to a Table of Contents and consists of the Topics defined for the documents in the library. A hypermedia author[2] specifies Topics within the original documents and assigns them a hierarchical order. After finding a Topic of interest, the reader selects it and the associated part of the document appears, as shown in **Figure 2.**

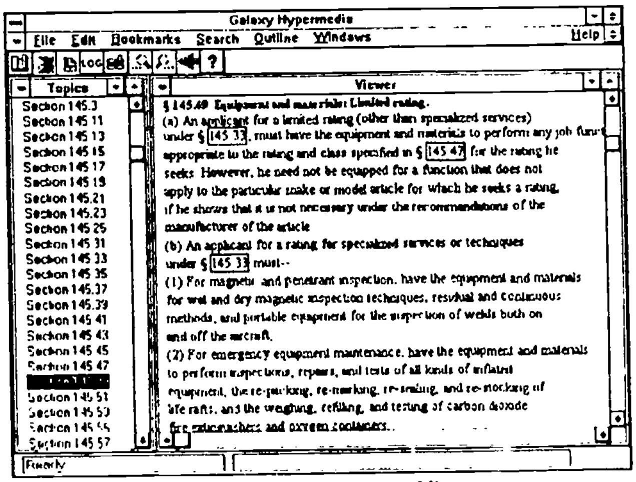


Figure 2 The Outline and Document Viewers

Hot Links. The HIS supports a variety of Hot Links a reader uses to navigate through the library. These include both inter- and intra-document links to text, links to graphics, Media Control Interface (MCI) supported media, definitions, and other executable programs. Hot Links are marked by a rectangular box surrounding red text (**Figure 2**).

Searching. One of the most powerful features of a hypermedia system is its ability to quickly locate specific words in large amounts of text without the reader having to scan each line of text. A reader performs a search by typing a query, as shown in **Figure 3**. The HIS then rapidly searches all documents in the library. When the search is complete, the HIS displays a list of the Topics satisfying the query, also shown in **Figure 3**. From that point, the reader can select one of the Topics for viewing. When the selected Topic's text is loaded, the search hits are highlighted, as shown in **Figure 4**. To see other search hits, the reader can scroll or use menu items.

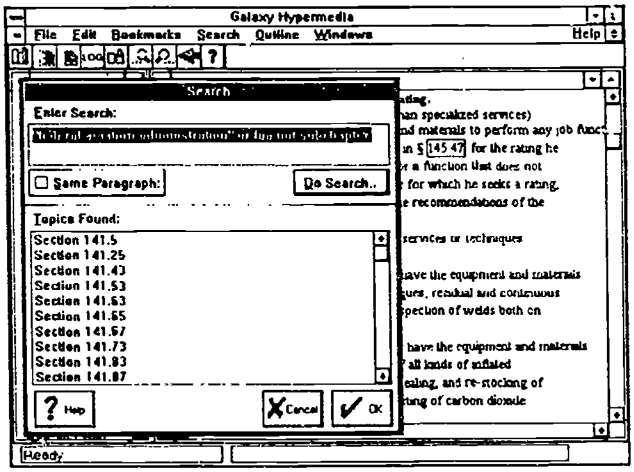


Figure 3 Search Query Dialogue Box

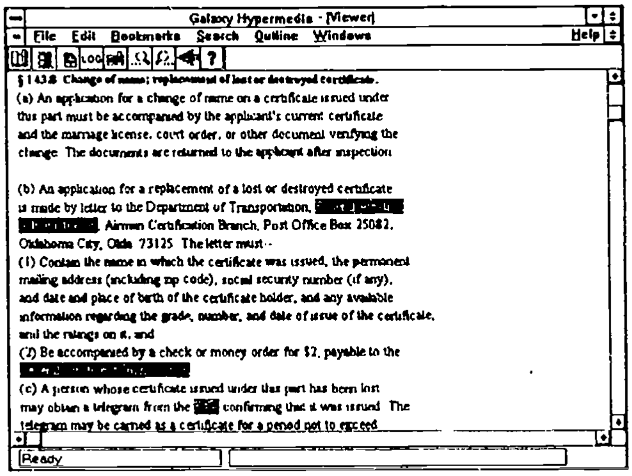


Figure 4 Search Hits

The HIS supports four types of searching: term, wildcard, phrase, and Boolean. The term search is the simplest. This is a search for one specific term, such as *aviation*, that is not a stopword. A stopword occurs so frequently in the document that it is not important; examples include *the* and *and*. Every Topic containing the search term is listed in the Search Query Dialogue Box.

The wildcard search allows the reader to look for variations of a term, i.e. *administrate*, *administration*, *administer*. A term or partial term can be appended with either an asterisk (*) or a question mark (?) wildcard. The asterisk represents zero or more characters, and the question mark represents zero characters or one character.

Phrase searching enables the reader to specify the order and adjacency of multiple search terms. For example, a phrase search for *"federal aviation administration"* will only display places where that exact phrase appears. The reader specifies a phrase search by placing quotes around the phrase to be searched.

Finally, the Boolean search combines any/all of the above types with Boolean operators (AND, OR, NOT), as in *"federal aviation administration" or faa not subchapter*. In this example, the search would return a list of all Topics containing either the phrase *federal aviation administration* or the term *faa*, but which do not contain the term *subchapter*.

Bookmarks. It is sometimes desirable for a reader to mark a place in a document. The HIS provides a bookmarking capability, enabling a reader to create multiple Bookmarks for a document. When creating a Bookmark, the HIS uses the Topic currently being viewed as the Bookmark's target destination. To use a previously created Bookmark, the reader chooses one from the list of active Bookmarks. The Topic containing the Bookmark does not have to be in the current library; the HIS will automatically switch libraries, if necessary.

Viewing

The HIS provides three distinct tools for accessing the various media comprising a hypermedia library. The Document Viewer has multiple entry mechanisms -- the Outline Viewer, the Search Query Dialogue Box, Bookmarks, and Hot Links. The Graphics Viewer and Multimedia Viewer are accessible only through Hot Links.

The Document Viewer. The Document Viewer, shown in **Figures 2** and **4**, allows a reader to scroll through and read a hypermedia document, as well as to investigate search hits. Text formatting such as boldface, italics, underlining, and multiple font size and face, enables the on-line document to closely resemble an original.

The Graphics Viewer. The Graphics Viewer appears when the reader clicks on a Hot Word linked to a static graphics image. It is used to view and print graphics. Supported graphics formats include, among others, bitmap (BMP), encapsulated postscript (EPS), target image file format (TIFF), and Joint Photographic Experts Group (JPEG). This is a seamless incorporation; the Graphics Viewer determines the format of the graphic and displays it appropriately.

The Multimedia Viewer. More innovative types of media are now available for computer presentation, e.g. sound, video, animation, etc. The Multimedia Viewer accommodates such media, determining the type of media when the reader selects a Hot Link to a media source and playing it appropriately. The HIS currently supports all MCI-supported media, including animation, video, cd-audio, and audio-video interleave.

Information Storage

Because a hypermedia document is more than just a digital version of a paper document, it is necessary to transform the original to a form that contains information for the HIS. This information runs the gamut from basic text formatting, such as which font to use, to links to other documents, graphics, animation, and other software programs. This transformation process (information storage shown in **Figure 5**) employs of a set of tools including editors, indexers, and database manipulators.

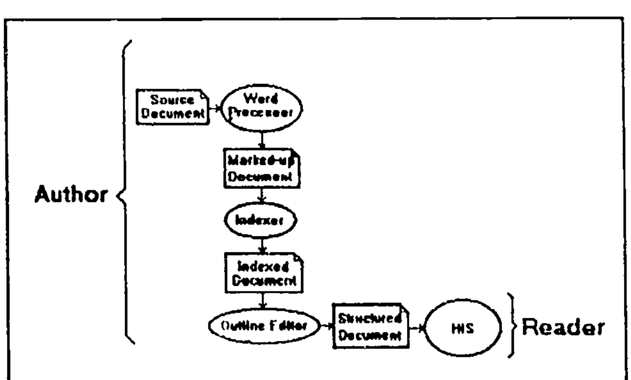


Figure 5 Information Storage Process

The first step in transforming a document is to create it or to open it in WordPerfect. This makes it possible to automatically take advantage of WordPerfect formatting such as boldface, italics, font, headers, etc. The author then adds Topic and Hot Link information to this document, using WordPerfect footnotes. The author also redlines any text which is to be designated a Hot Link in the viewer. This modified document is called a **Marked-up Document**. At this point, the marked-up document is ready to be indexed.

The Indexer is a two-pass tool. The first pass filters the WordPerfect document, translating it to ANSI text. The WordPerfect formatting information mentioned above is automatically translated into GML, a proprietary language developed by Galaxy, in the ANSI file. The footnote information the author added is also translated into GML. During the second pass, the Indexer parses the document and stores all information in a database. This is referred to as an **Indexed Document** in **Figure 5**. Each line of text is stored in the database, along with any GML formatting, Topic, and Hot Link information for that line. This makes document display and scrolling rapid and accurate. It also makes phrase searching precise. Each non-stopword in the document is also stored, along with its location in the document. This ensures quick, accurate term and wildcard searching. The database format is designed for optimal retrieval performance in large documents. A database storage mechanism increases speed in displaying a document, switching between documents, and searching. The database management system (DBMS) being used is Raima Corporation's Raima Data Manager which is compatible across PC, UNIX, Macintosh, and other platforms.

After a document is indexed, it must be structured. This step specifies the relative ordering, as well as hierarchical ordering, of all Topics the author defined in the WordPerfect document and results in the **Structured Document**. The author orders the Topics using the Outline Editor. The Outline Editor is similar to the Outline Viewer in the HIS reader interface; however, Topics are read-only in the Outline Viewer. In the Outline Editor, they are modifiable. Document structuring modifies the database. A collection of one or more structured documents forms a library to be viewed in the HIS.

HIS Applications

Now that the HIS has been described in detail, it is beneficial to how it has been used. As mentioned above, the driving force behind this project is the aviation industry. Aviation maintenance technicians, depending on their experience and maintenance role, estimate that as much as forty percent of their workday can be consumed with accessing technical information, usually spanning many volumes. Trying to locate all references to a particular component or procedure in such a large collection of data is daunting and time-consuming. It would require hours of effort, with no guarantee of locating all the references. Initial studies have found that the time spent searching a maintenance manual is reduced by as much as forty percent when the manual is on CD-ROM instead of paper or microfiche (Cruickshank, 1993). Improvements in the way aviation maintenance personnel access information will lead to more reliable and cost-effective aircraft maintenance. Toward this end, the HIS has been successfully integrated into multiple tutoring and job aiding systems for the FAA. These are described below.

The Environmental Control System (ECS) Tutor. This software (described in FAA/AAM & GSC, 1993) was sponsored by the FAA Office of Aviation Medicine as part of the HFAM research program. The ECS Tutor is an intelligent tutoring system which simulates the operation of the Air Conditioning portion of the ECS for the B-767-300.

Training manuals for the Boeing 767's ECS were indexed for access via the HIS. During a tutoring session, when a student requests information about ECS parts and procedures, it is displayed using the HIS. This allows direct access to information in a format familiar to the students using the tutor.

The Air Traffic Control Beacon Interrogator (ATCBI) Tutor. This software was sponsored by the FAA Technical Center, Advanced System Technology Branch (ACD-350) and is described in Jones and Jackson, 1992. This proof-of-concept tutor investigates the use of advanced technology for Airways Facilities maintenance training. The goal of the training system is to help experienced technicians maintain proficiency of their knowledge about the ATCBI-4.

This tutor contains a simulation of the ATCBI-4 and references information about the ATCBI-4. When requested by the student, information from the technical manual is brought up in the HIS. This information includes part/output descriptions, test/adjustment/replacement explanations, preventive maintenance procedures, standard and tolerance values, and functional block schematics.

The Performance Enhancement System. The Performance Enhancement System (PENS) (described in FAA/AAM & GSC, 1993) applies pen computer and hypermedia technology to provide real-time job aiding and information retrieval for Aviation Safety Inspectors (ASIs) who must have access to large amounts of information. The Federal Aviation Regulations and the Airworthiness Inspector's Handbook have been indexed and put into a library for inspectors' use. As inspectors use PENS, they can directly access the HIS to reference and search for information. PENS is initially being distributed to ASIs in 9 U.S. locations for use and evaluation. Feedback will be provided regarding the HIS and will be used to make future enhancements.

Summary

The HIS has proven that hypermedia technology is useful to the aviation industry. For the HIS to continue to succeed, it must continue to evolve. The HIS must expand its document base and adhere to aviation documentation standards, such as Air Transport Association Specification 100 and Standard Generalized Markup Language (SGML). Also, because commercial and military aviation have many similarities, the HIS must also consider Department of Defense (DoD) standards. The DoD is mandated to use SGML by the Federal Computer-aided Acquisition and Logistical Support (CALS) program. Also, new weapons systems use Interactive Electronic Technical Manual (IETM) Specifications. The HIS will consider these and other emerging standards.

Additionally, in order to facilitate the creation of hypermedia documents, future work will improve the support tools for the author. The research will investigate ways to make the conversion from existing source documents to their hypermedia counterparts less time-consuming. This will include the development of filters for other word processing packages.

Advanced technology information systems such as the HIS will save the aviation industry time and money (Johnson, 1993). In addition, by providing a complete search of information, the HIS supports safer maintenance. While the HIS is only a part of safer, more cost-effective maintenance, it is an important part. Continued, enthusiastic cooperation from the aviation industry makes this possible.

Acknowledgements

This work has been supported by the Federal Aviation Administration Office of Aviation Medicine under contract DTFA01-92-Y-01005.

References

Cruickshank, D. (Spring 1993). Moving mountains at Boeing. *Iris Universe*, pp. 12-17.

Federal Aviation Administration, Office of Aviation Medicine (FAA/AAM) & Galaxy Scientific Corporation (GSC). (1993). *Human factors in aviation maintenance - Phase three, volume one progress report.* (NTIS No. DOT/FAA/AM-93/15).

Federal Aviation Administration, Office of Aviation Medicine (FAA/AAM) & Galaxy Scientific Corporation (GSC). (1994). *Human factors in aviation maintenance - Phase three, volume two progress report.*

Johnson, W. B. (1993). Human factors research: Can it have an impact on a financially troubled aviation industry? *Proceedings of the 37th Annual Meeting of the Human Factors and Ergonomics Society.* Seattle, WA: The Human Factors and Ergonomics Society.

Johnson, W. B. & Norton, J. E. (1992). Integrated information for maintenance training, aiding, and on-line documentation. *Proceedings 36th Annual Meeting of the Human Factors Society.* Atlanta, GA: The Human Factors Society.

Jones, J. A. & Jackson, J. (1992). Proficiency training systems for airway facilities technicians. *Proceedings 7th Annual Meeting on Human Factors Issues in Aircraft Maintenance and Inspection.* Atlanta, GA.

[1] *Reader* refers to someone using the HIS to view documents.

[2] *Author* refers to someone who puts a document or collection of documents into a hypermedia library for use by the HIS.

JUGAME: Game Style ICAI System for Kanji Idiom Learning

Toshihiro Hayashi and Yoneo Yano
The University of Tokushima
2-1, Minamijosanjima, Tokushima 770 Japan
E-Mail: hayashi@is.tokushima-u.ac.jp

Abstract: In learning the Japanese language, kanji idiom learning is as important as kanji(Chinese characters) learning. We have developed a kanji idiom learning ICAI System called JUGAME. It can handle approximately 3,000 kanji idioms and has a puzzle game environment within a learning environment using OSF/Motif graphic user interface. In this environment, it is the goal of the game for student to find all kanji idioms hidden in the puzzle by combining puzzle elements(kanji). Although it is generally said to be difficult for the foreigners to learn kanji idioms, they can learn kanji idioms through playing the puzzle game without losing their motivation. Moreover, JUGAME does not prepare the puzzle patterns in advance. According to the teaching strategy, JUGAME generates an adaptive puzzle pattern focusing on the knowledge level of the student. We have implemented JUGAME on SONY EWS NWS-1750.

Recently, the number of foreigners who are learning the Japanese language has in creased. Kanji learning is one of the most difficult parts of learning the Japanese language for foreigners in Japan, and they need a good educational environment for kanji learning. For this social need, many electronic dictionaries(Nakajima, 1988; Hayashi, & Yano, 1991; Bhatia, 1992; Walters, Fahy, Nakamura, & Reid, 1992) and kanji learning CAI applications (Yamasaki, Yamamoto, & Inokuchi, 1990; Hayashi, & Yano. 1993) have been developed. We have developed a kanji idiom learning ICAI system with a puzzle game environment called JUGAME(Yano, Miyoshi, & Hayashi, 1993) under the guideline of environmental ICAI system and it consists of a domain knowledge base for kanji idioms, a learning environment, a puzzle generator(teaching control module) and an advisor(Fig. 1). The educational aim of JUGAME is for foreigner to learn kanji idioms and their structure, meanings, pronunciations and the related idioms. In addition, the instructional domain of JUGAME is made up of kanji idioms constructed using only two kanji.

JUGAME provides the student with a kanji puzzle game as a learning environment. In this environment, the student tries to construct kanji idioms from a combination of puzzle elements(kanji) and s/he can learn kanji idioms themselves through the puzzle game. Moreover, the student can learn about the constructed kanji idioms from a dictionary function of JUGAME. The patterns of the kanji puzzles are not prepared in advance but JUGAME automatically generates a kanji puzzle. The hidden kanji idioms are appropriate for the knowledge level of the student because the teaching control module selects them based on the teaching strategy. In addition, the advisor function helps the student avoid hitting a learning plateaus in the learning environment, but this paper omits its detail (Please refer to the paper(Yano, Miyoshi, & Hayashi, 1993) about advisor function).

Domain Knowledge Base

Knowledge Representation of Kanji Idiom

We have classified kanji idiom knowledge into the kanji idiom structure, the kanji attributes and the relational words. We call kanji idiom structure, "SIW"(Structure of Idiomatic Words), kanji idiom attributes, "UIW"(Usage of Idiomatic Words) and relational words, "RIW"(Relation between Idiomatic

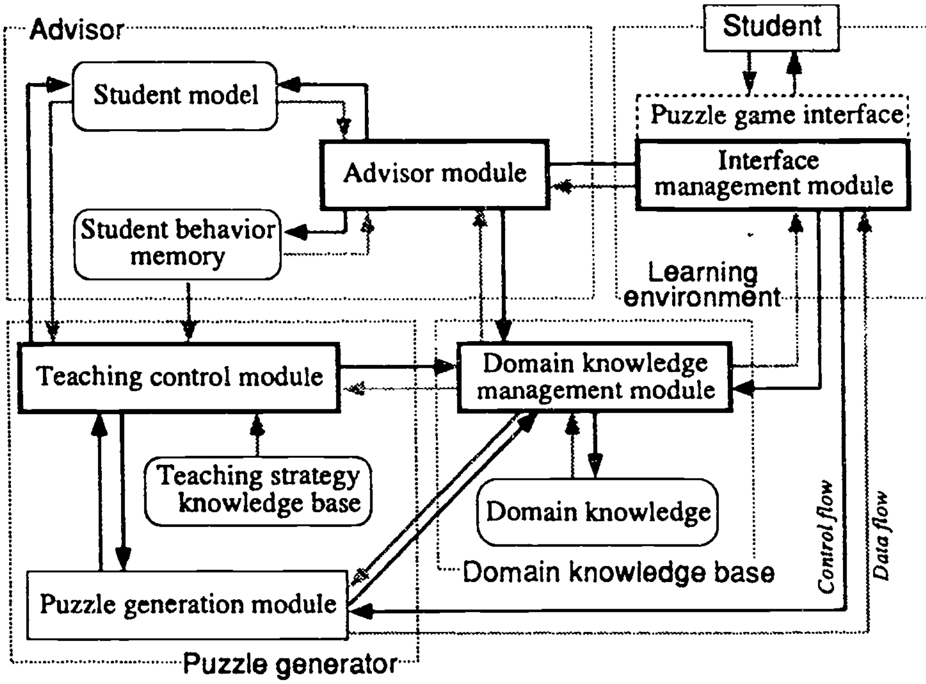


Figure 1 System Configuration

Words). SIW puts emphasis on the relationship between a kanji idiom and its kanji components. SIW represents the kanji components, their meanings, and their pronunciations. In addition, SIW represents the relationship between the two kanji components. UIW puts emphasis on the attributes of a kanji idiom. UIW consists of the meanings, the pronunciation and so on of the kanji idiom. RIW puts emphasis on the relational idioms, such as homonym, synonym and antonym, of a kanji idiom. We do not represent RIW explicitly because RIW is inferred from the attributes of kanji idioms in most cases.

Table 1 Triplet Representation of Kanji Idiom

	Triplet representation	*Example*
SIW	(<idiom>, CONSTRUCTION, <kanji>)	(牛乳, CONSTRUCTION, 牛) (牛乳, CONSTRUCTION, 乳)
	(<idiom>, <kanji>, <meaning of kanji>)	(牛乳, 牛, cow), (牛乳, 乳, milk)
UIW	(<idiom>, READING, <reading>)	(牛乳, READING, gyuunyuu)
	(<idiom>, MEANING, <meaning>)	(牛乳, MEANING, cow's milk)
	(<idiom>, CONCEPTION, <conception>)	(牛乳, CONCEPTION, drink)

Implementation of the Knowledge Base

For implementation of the knowledge base for kanji idioms on a computer, the kanji idiom knowledge is represented by triplets Triplets consist of an entity, an attribute and a value, formalized as (e, a, v) = (entity, attribute, value). Attributes refers entities to values. Any triplet can be used as search conditions in the knowledge base module. The elements e, a, v of a triplet can have '*' as a meta-character. '*'

matches with all strings. For example. (*E*, *A*, *) matches with all triplets of which the entity is *E* and the attribute is *A*. We call these searches "asterisk searches". We represent kanji idioms' data in triplet format. Table 1 shows an example of the triplet representation of "牛乳."

Searching of Kanji Idiom Knowledge

In the knowledge base, SIW and UIW are searched by asterisk searches but RIW cannot be searched the same way because the kanji idioms do not link directly to their relational kanji idioms. Therefore, JUGAME has some searching procedures to find the relational kanji idioms. With regard to RIW search, for example, a synonym search finds some synonyms using the combination of the asterisk searches. For example, if the search condition is 以外, then JUGAME finds (以外, READING, igai) using an asterisk search with (以外, READING, *). Then JUGAME finds (意外, READING, igai) using an asterisk search with (*, READING, igai). In this way, JUGAME can find the synonyms.

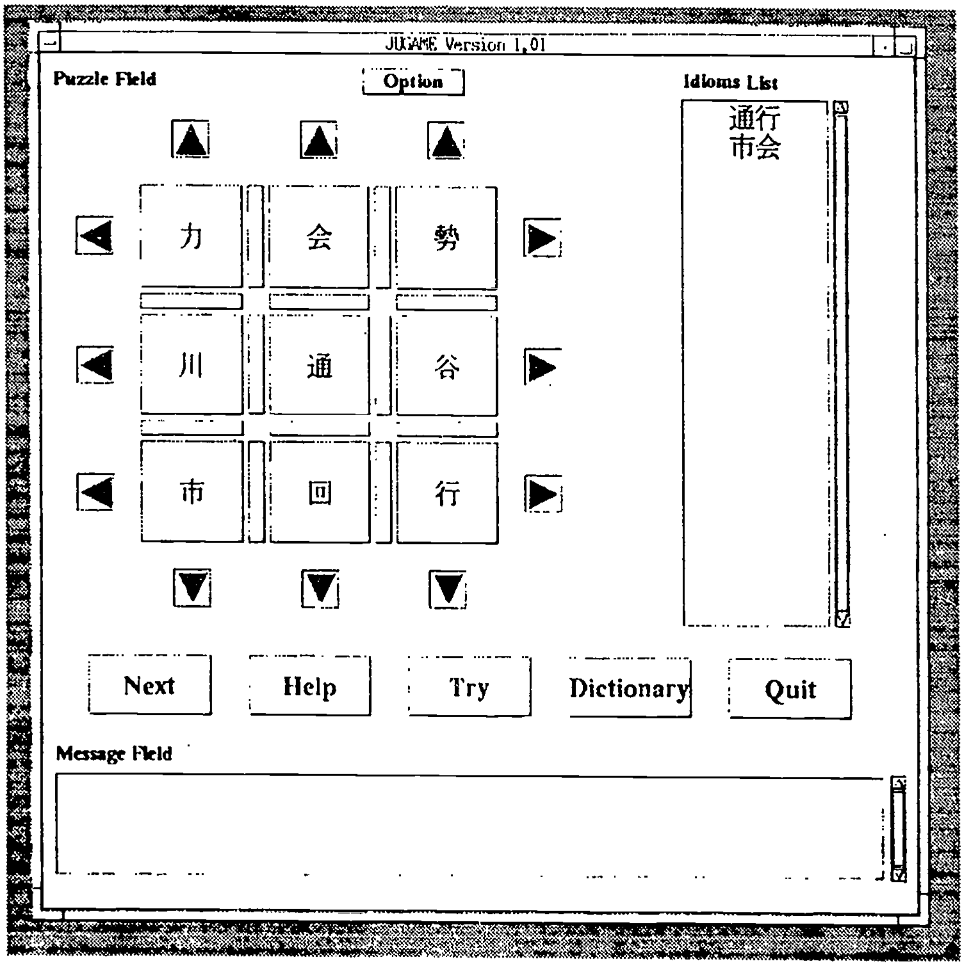


Figure 2 Learning Environment of JUGAME

Learning Environment

JUGAME gives an appropriate learning environment for students who have already learned several kanji. In the learning environment, the students can learn the following: (1) The structure of kanji idioms; (2) The attributes, such as meaning and pronunciation, of kanji idioms; and (3) The relational idioms, such as synonyms, of kanji idioms. The student can learn the above knowledge indirectly, while s/he solves the kanji idiom puzzles in the learning environment. We think that playing the kanji puzzle game will motivate the student into learning new kanji idioms.

Configuration of the Learning Environment

Fig. 2 shows the learning environment. The learning environment consists of a puzzle field, a kanji idiom field, a message field and puzzle operation buttons.

JUGAME generates kanji puzzle patterns automatically. The kanji puzzle consists of nine kanji placed in 3 by 3 matrix. There are nine buttons called shift buttons around the matrix. The student can perform some of the operations of each row and each column of the kanji puzzle by clicking on a shift button. The student can rotate a row of three kanji to the right or to the left by clicking on a shift button. The student can also move a kanji up or down by a similar column operation.

There are twelve buttons called check buttons between each adjoined kanji. If a student thinks this seems to be a kanji idiom focusing on a pair of two kanji after some row and column operations, s/he can choose the pair by clicking on the check button between the two kanji. The two kanji change color highlight this indication. The student can also choose other pairs of kanji this way. After some of the above operations, the student can check whether the kanji pair becomes a kanji idiom or not by clicking on the try button. After clicking on the try button, JUGAME searches the kanji idiom constructed by the kanji pair. If JUGAME can find the kanji idiom, it shows and stores the kanji idiom in the kanji idiom field. On the other hand, if JUGAME cannot find the kanji idiom, it prints a comment such as, "No kanji idiom is constructed by this kanji pair" in the message field. If the student can find no kanji idiom in the kanji puzzle, the student can learn some unknown kanji idioms in the kanji puzzle by clicking on the help button. The student can construct kanji idioms through easy these button operations.

Illustrated Play of Puzzle Game

The student can learn kanji idioms by going through the above operations. Fig. 3 shows an example of the kanji puzzle operation. Example A shows the construction process of kanji idiom 人権. Example B shows the other construction process of the kanji idiom 人権. In addition, it happens that 達人 is constructed at the same time in example B.

Puzzle Generator

Selection of Puzzle Element

The student is allowed to learn freely in an environmental CAI system. If the learning control by the system is implicit, it does not prevent her/him from learning freely. We adopt the implicit teaching strategy for JUGAME. When JUGAME generates a puzzle, it can control student's action implicitly. JUGAME has three teaching processes: (1) the basic knowledge teaching process; (2) the knowledge stability diagnosis process; and (3) the relational knowledge teaching process. The basic knowledge teaching process is to teach kanji idioms themselves, the pronunciations, the meanings and so on. If the student can construct a kanji idiom on the puzzle game, JUGAME assumes that either the student knows the kanji idiom, or it is unknown to her/him and the student has guessed. If the student did not construct a kanji idiom from puzzle, it is continuously adopted as a puzzle element in the following puzzles. The knowledge stability diagnosis process is to diagnose whether a kanji idiom which is passed from process (1) is stabilized knowledge or not. Based on a hyposis that *a student can use stabilized knowledge for problem solving even if it has been long time since s/he learned of the knowledge*, this process is executed over a long interval. If the student constructs the kanji idiom again in this process, JUGAME assumes that the kanji idiom is stabilized knowledge. If the student cannot construct the idiom again JUGAME assumes the student guessed and the idiom is sent back to process (1). The relational knowledge teaching process is to teach some relational knowledge of the stabilized kanji idioms. In this process, the elements of the puzzle are relational kanji idioms. Although there are many types of relational kanji idioms, this process selects the proper types and adopts some kanji idioms of these types as puzzle elements.

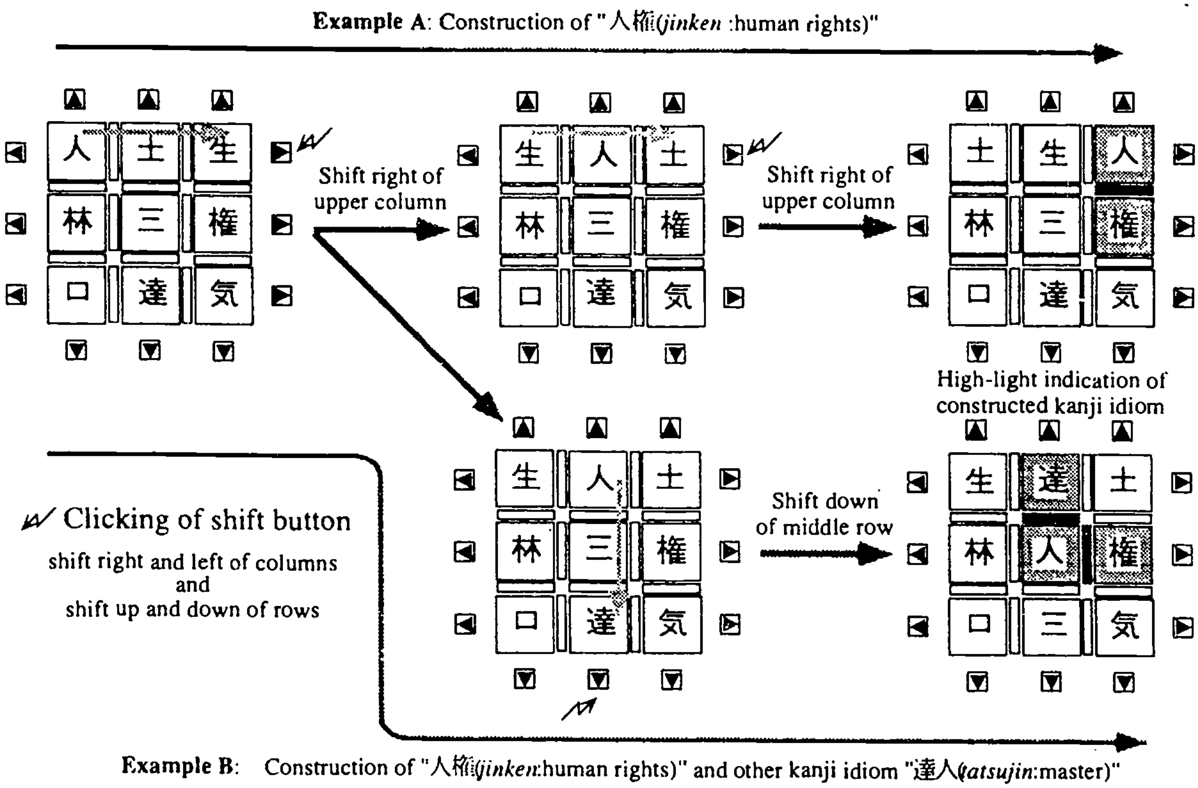


Figure 3 Illustration of Game Progression

Parallel Execution of Teaching Process

In ITSs a process for teaching knowledge cannot run until the previous process for teaching other knowledge has finished. On the other hand, JUGAME has the potential to teach many kanji idioms during one puzzle game. Using this feature, JUGAME can execute different processes for teaching many idioms at the same time. For example, JUGAME can execute the basic knowledge teaching process for idiom A, the knowledge stability diagnosis process for idiom B and the relational knowledge teaching process for idioms C and D during one puzzle game. By the parallel execution of these three processes, JUGAME can teach many kanji idioms effectively. For the implementation of the processes, we adopt multiple agents as respective teaching processes. The agents adjust each request by their message passing and select the elements for the new puzzle. Fig. 4 shows the teaching processes and the respective agents.

Puzzle Generation

JUGAME generates the kanji puzzle automatically using the following processes: First it selects four or less kanji idioms based on the teaching strategy; Then it separates each kanji idiom into two kanji and gathers the kanji into a kanji list.; Then it deletes overlapped kanji from the kanji list and adjusts the kanji list to have nine kanji by adding some random kanji into the kanji list; Then it picks out all kanji idioms constructed by the kanji pairs in the kanji list, and stores the kanji idioms into an idiom list; It generates a kanji puzzle which consists of the nine kanji and checks whether the kanji idiom is already constructed or not in the puzzle using the idiom list. If it finds the constructed idiom, then it performs the previous process again. If it cannot find the kanji idiom, then it performs the next process. Finally, it puts the generated puzzle in the puzzle field. For example, in the case of Fig. 4, JUGAME selects four kanji idioms such as "人権", "人気", "人生", and "人口" based on the teaching at first. These kanji idioms contain '人'. Next, it extracts '人', '権', '気', '生' and '口' from the four kanji idioms and store the five kanji into a kanji list. Then, it adds '林', '土', '三' and '達' into the kanji list. Finally, it generates a

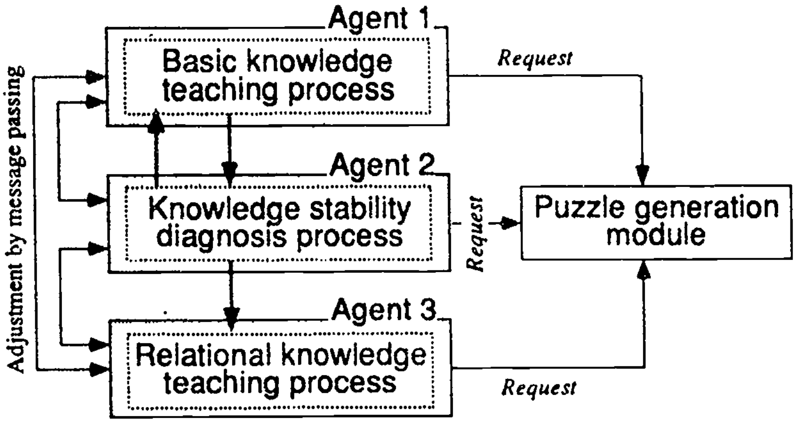


Figure 4 Teaching Processes and Their Agents

kanji puzzle pattern which includes the nine kanji.

Conclusion

In this paper, we described an environmental ICAI system called JUGAME which has a puzzle game environment. In this environment a student tries to solve a puzzle by constructing kanji idioms using the puzzle elements(kanji). The student's actions lead her/him to learn kanji idioms. We think that JUGAME gives a new and motivated learning environment for foreigners to learn kanji idioms. Moreover, JUGAME can control the generation of puzzle patterns implicitly using the teaching control module. This module can select appropriate kanji idioms hidden in the puzzle by referring to her/his knowledge stability.

We have developed JUGAME using C language and OSF/Motif GUI library (a part of JUGAME is coded by Xlib level programming) on SONY EWS NWS-1750. The knowledge base of JUGAME stores 2,826 kanji idioms and they are represented by approximately 35,000 triplets. In spite of many triplets, JUGAME can search through information and generate appropriate puzzle patterns quickly.

Reference

Bhatia, A. (1992). Kanji retrieval by recursive location of elements using hypercard. *CALICO Journal*, 9(4), 4-25.

Hayashi, T. & Yano, Y. (1991). Development of a Kanji Dictionary Focusing on the Method of Constructing Japanese Kanji. In *Proceeding of International Conference on Multi-Media in Education and Training*, 121-126.

Hayashi, T. & Yano, Y. (1993). Development of an Environmental ICAI System for Kanji Learning. *Proceeding of International Conference on Computers in Education*, 132-137.

Nakajima, K. (1988). On developing hypercard stacks for the study of Chinese Characters: KANJI CARD. *CALICO Journal*, 6(2), 75-87.

Yamasaki, T., Yamamoto, M. & Inokuchi, S. (1990). Computer Coaching for Beautiful Handwriting of Japanese Characters in Elementary School. In *Proceeding of 5th World Confernce on Computer in Education*, 725-728.

Yano, Y., Miyoshi, K. & Hayashi, T. (1993). Development of a Game Style Environmental ICAI System for Kanji Idioms Learning. In *Proceeding of International Conference on Computers in Education*, 411-413.

Walters R. F., Fahy D., Nakamura A. Y., & Reid C. M. (1992). Kanji-Lookup: A Computer-Based, Multi-Indexed System for Beginning Students of Japanese. *Journal of Computer-Based Instruction*, 19(1), 27-32.

The Intelligent Workbook

T. HEIFT and P. MCFETRIDGE
Department of Linguistics
Simon Fraser University, Burnaby, B.C. Canada V5A 1S6
E-mail: heift@sfu.ca mcfet@cs.sfu.ca

Abstract: Burghardt (1984) defines an authoring system as a computer program which eliminates the need for course instructors to learn computer programming but instead constructs pedagogical software based on the instructors responses to questions on teaching strategy and course material. In this paper we will outline an authoring system which not only exhibits this ease of use and flexibility in its design, but demonstrates intelligent behavior, one of the significant interactive qualities of Computer-Assisted Language Learning. Intelligence is simulated by daemons each of which seeks a specific error in student input, providing the student with immediate error-contingent feedback. The model, which allows the course designer to create any type of language exercise is demonstrated with German as the target language and English as the source language.

Introduction

All compelling arguments for Computer-Assisted Language Learning (CALL) assume intelligent responses to students' input. Without intelligence the computer is merely another medium for presenting information, one not especially preferable to a static medium such as print. In order to go beyond the multiple choice questions, relatively uninformative answer keys and gross mainstreaming of students characteristic of workbooks, a model should emulate significant aspects of a student-teacher interaction.

This paper presents an authoring system for CALL. The system is designed to produce an intelligent tutor, rather than an electronic workbook: the principal advantage and contradistinction being that an intelligent tutoring system provides the student with evaluative or error-contingent feedback (Alessi and Trollip, 1985). Venezky & Osin (1991) stress that "for almost all cognitive learning, instruction is enhanced by evaluative feedback. In many cases it is essential, if any learning is to occur. Translation of a foreign language is a prime example of the latter situation (pg. 9)."

Turning to a concrete example of error-contingent feedback, consider the following grammar exercise intended to practice word order in subordinate clauses in German. In the main clause the verb occurs in second position; however in a subordinate clause the verb must be in sentence-final position. The student is presented with the exercise:

(1) Build a sentence with the words provided:
Ich / nehmen / zwei / Menüs / weil / ich / haben / Hunger.

The simplest system merely reports an error if the student does not supply the correct answer, "Ich nehme zwei Menüs, weil ich Hunger **habe**", perhaps revealing the correct answer after a number of tries. By comparison, error-contingent feedback responds with a description of the error, and performs a deeper linguistic analysis in order to isolate the source of the error. If the student responds with "Ich nehme zwei Menüs, weil ich ***habe** Hunger, the system responds with "incorrect word order in subordinate clause". The more detailed error analysis guides the student toward the correct answer, and provides an invaluable record of the student's performance for later evaluation and remedial work.

The error-contingent feedback which serves as the basis for effective teaching in the language classroom can be transferred to the computer. This has been previously demonstrated with an intelligent tutoring system on a small micro computer platform teaching subordinate clauses in German (Heift 1993, Heift & McFetridge 1993). In this project, all error checking routines — which we have called daemons — were built rather painstakingly. This process is time consuming and requires expert knowledge of programming as well as second language instruction. The project we report here is an authoring system for creating intelligent workbooks. It provides a method for specifying which daemons are required for each exercise and how they are parameterized. The computer code which analyses student errors is automatically generated.

The Daemon Approach

Error-contingent feedback is provided through the use of daemons which only seek errors relevant to the exercise. A daemon is a program submodule, typically highly parameterized, which seeks a particular error and takes remedial action when that error is discovered. Daemons are conceptually simple but can simulate intelligence. They achieve this, in part, because the pedagogical principles of 'selection' and 'gradation' constrain the problem domain— the range of possible errors in any given exercise is small, or at least finite, compared to many natural language processing tasks. They are simple in the sense that each daemon is responsible for checking only one type of error— in our example above, for instance, one daemon would test solely for word order, while separate daemons would each check verb inflection, spelling, etc. This modularity allows the programmer to create a pool of daemons for each exercise, adding daemons from previous exercises to the current one. The daemon approach ensures relevance of response— the order in which daemons are activated keeps the point of the current exercise salient. Additionally, the overall system can easily be extended to encompass new phenomena by adding new daemons to the pool.

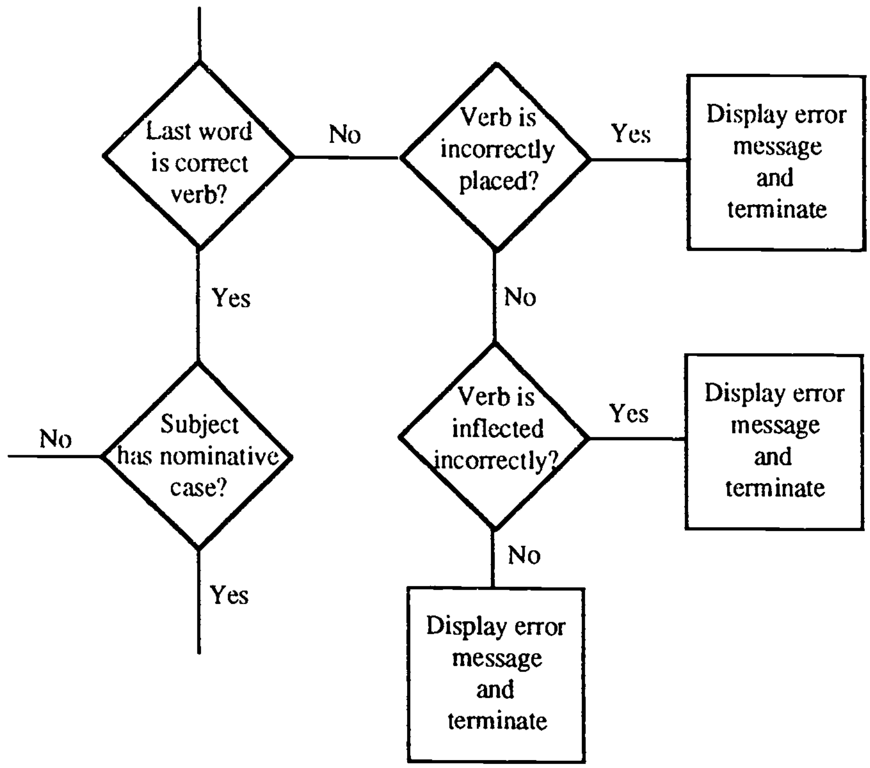
The daemon approach also allows us to make efficient use of a small computer platform, which in turn improves the student/program interaction. Generally, the amount of knowledge and processing time required by a tutoring program increases with the sophistication of the error analysis. We can, however, establish an approximate hierarchy which reflects the kinds of errors most likely to occur, and call the daemons in that order. The hierarchy itself is based on what has been taught most recently, and to some extent, on a contrastive analysis between the native/target language of the student. An additional consideration might be the complexity of the grammatical construction. We can reasonably assume that given two constructs of different complexity, the more difficult of the two will result in more frequent and more persistent errors.

Conceptually powerful and computationally efficient, the daemon approach offers still another attraction: reusability. In the same way that daemons can be conjoined and brought to bear on a series of exercises, much of the research and development time spent in creating one tutoring program can be applied to the next. The rationale underlying an intelligent tutor's design then provides the framework for a higher-level authoring system.

Our goal is to produce tutoring systems which provide intelligent and informative feedback to students' errors. Ours is a more ambitious project than many on two parameters. First, we are attempting to open up the domain of errors which the system can analyze. The simplest system associates messages with particular responses and constrains the student to select one. We wish to allow students freedom in their responses and consequently require greater generality in the routines which analyze the students's input. Second, the system we present here is not a tutoring system itself; it is a workbench for creating a tutoring system. We have decomposed the process of error detection into a small set of subroutines which the program writer can use to create daemons and to build exercises. The resulting tutoring system is one we call an *intelligent workbook* because it has the superficial appearance of an electronic workbook, but incorporates sophisticated error analysis and feedback.

We illustrate how a set of daemons is constructed by first considering the problem of checking that the verb in the example above is inflected correctly and is in the correct position. The portion of a flow chart for an exercise relevant to this task is given in figure 1. Each decision point in the flow chart corresponds to a daemon. The first daemon checks that the final word in the sentence is the correctly inflected verb. It functions as a gateway to other daemons: if the last word in the sentence is not the expected word, the flow of control is passed to daemons which determine what error has been made. The first of these scans the sentence for the verb and if successful displays an error message informing the student that the verb is not in the correct position. It terminates the program to give the student an opportunity to correct the mistake. The second daemon scans the dictionary entry of the verb for an inflected form corresponding to that which the student entered. If it finds one, then the verb must be incorrectly inflected and an error message reporting this is displayed. Finally, if the student's error is so severe that neither daemon can successfully analyze it, a generic error message is displayed.

Figure 1



A System Overview

The task of the program writer is to convert the flow chart in figure 1 into a program. The system we describe here is a workbench which modularizes the construction of a program and presents the writer with a set of packaged daemons. Creating an intelligent workbook is a process of several well-defined stages.

In the first stage, the set of daemons that will be required is defined. We describe this process in the section on daemon functionality. We view this process as iterative and cumulative, and consequently have provided editors which allow the program writer to amend the definitions of daemons. It is also possible to add new daemons to the pool at any time. As a consequence, it is possible to begin with a workbook which can respond to only the grossest errors. Over time, the program writer can build greater functionality into the workbook without changing its overall structure.

After the pool has been defined, the writer defines the exercises which comprise the workbook. As in the print medium, an exercise has a set of instructions and requests input from the student. Part of the definition of a exercise is the set of daemons which are assigned to the exercise. When a daemon is assigned to an exercise, its *parameters* must also be assigned. For example, the task given to a student in example (1) above requires that the verb be placed at the end of the sentence. The daemon that first checks that the last word in the student's answer is the verb has two parameters: the correct answer and the position of the word in the student's answer which must correspond to the correct answer.

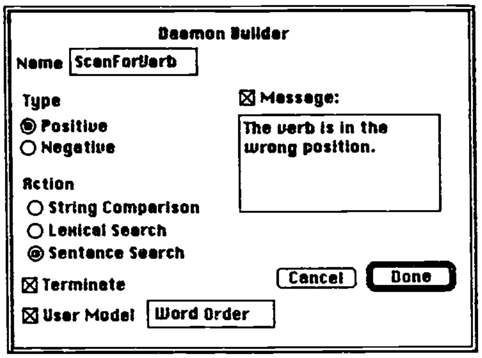
The final stage in building the workbook is defining the order in which the exercises are to appear and saving the workbook. The workbook is a stand alone application which presents the student with a series of exercises and applies the daemons to the students responses to each exercise.

Daemon Functionality

The flow chart in figure 1 demonstrates three kinds of error analyses. The first decision point compares two forms and takes action based on the success or failure of the comparison. The second decision point relies on a scan of the sentence. If the verb can be found somewhere in the sentence, the associated error message is displayed. The final decision point searches the lexicon for the form which the student entered. To this point of our analysis of error detection, this list exhausts the possibilities.

As an example, the definition of the daemon corresponding to the second decision point is illustrated in figure 2.

Figure 2



The user defines a daemon by providing a name and specifying the action which the daemon will perform. The actions correspond to three decision points in the example in figure 1:

(1) String Comparison

In a substring search daemons are directed to specific positions within the sentence. This applies to daemons which check word position or verb inflection, as in example (1). Further examples are daemons which check prepositions, inflection of attributive adjectives, gender, case, etc.

(2) Lexical Search

In a lexical search the daemons refer to a database, such as a glossary, or dictionary. For example, in the flow chart above the third decision point corresponds to an attempt to find the last word in the sentence in the dictionary under the lexical entry of **haben**. If it is found, then the student has placed the verb in the correct position but has inflected it incorrectly.

(3) Sentence Search

This kind of daemon searches the entire sentence for specific strings; for example, if a verb is not found in the correct position, scanning the entire sentence for the verb will establish whether the student has placed the verb incorrectly. The example in figure 2 illustrates the construction of a daemon of this type. As the flow chart in figure 1 illustrates, this daemon is activated only if it is previously established that the last string in the sentence is not a correctly inflected form of the verb. The ScanForVerb daemon will look through the sentence for the verb and display an error message if it successfully find the verb.

The *type* distinction is equivalent to the *yes/no* paths in the flow chart. A positive type will display an associated message if the specified action is successful. A negative type will display an associated message if the specified action is unsuccessful. In the flow chart in figure 1, the daemon corresponding to the first decision point will be negative. Its function is not to display a message but to serve as a gateway to the other daemons.

Two other properties of a daemon are determined by the state of the check boxes in the lower left. If the *terminate* box is checked, all further processing is stopped after the message is displayed. This feature gives the student an opportunity to correct the error before presentation of other errors. By default, all daemons which display a message terminate.

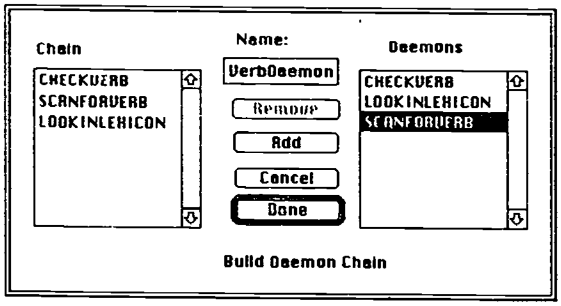
The *user model* check box establishes a count for the error tracked by the daemon. Each time a daemon successfully discovers an error, the count associated with it is incremented. This count can be displayed at the end of a session or be used to direct the student to remedial exercises.

Interrelation of Daemons

It is often necessary for daemons to cooperate in the analysis of a student's input. It is also useful to be able to refer to a large group of daemons by a single name; this is particularly true if the same set of daemons is used in a number of exercises. We have provided two different ways of combining daemons into larger program structures.

The simplest method is to simply group daemons into an ordered set. This set is named and can be applied to an exercise by name. It is also possible for daemons to be mutually exclusive. This situation arises when an error has been discovered, but there are many possible sources of the error. If a daemon successfully fires, it is not necessary to check the others. In figure 3, a chain of daemons with this latter property is constructed. The name of this daemon is VerbDaemon. It consists of three daemons corresponding to the three decision points in the flow chart in figure 1. The first checks that the verb is correct. If it is, the daemon signals success and the rest of the daemons in the chain are ignored. If the verb is incorrect, the other daemons are applied. The ScanForVerb daemon was defined in figure 2. The LookInLexicon daemon corresponds to the third decision point in the flow chart in figure 1.

Figure 3

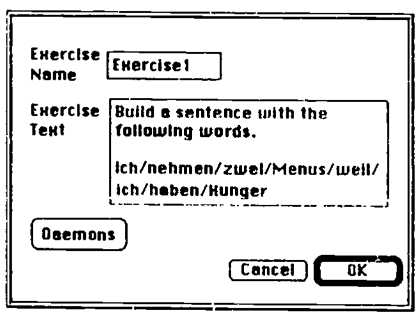


Groups of daemons which are ordered but not mutually exclusive can be built up similarly.

Defining an Exercise

The penultimate step in creating a workbook is setting the exercises. We provide a window in which the text of each exercise is specified. The exercise from the example in (1) above is defined in figure 4. The daemon VerbDaemon, defined in figure 3, is assigned to this exercise. At this point the system requests values for the parameters of each of the subdaemons. The CheckVerb daemon will be parameterized with *habe* and 8. That is, the answer is incorrect if the 8^{th} word is not *habe*. The ScanForVerb daemon will be parameterized with *habe*. This daemon will scan the student's input for this exercise and signal the associated error message if *habe* is found. Finally, the LookInLexicon daemon will be parameterized with *haben* and 8. It will search the lexical entry of *haben* for the word found in 8^{th} position of the input and signal the associated error message if it is found.

Figure 4

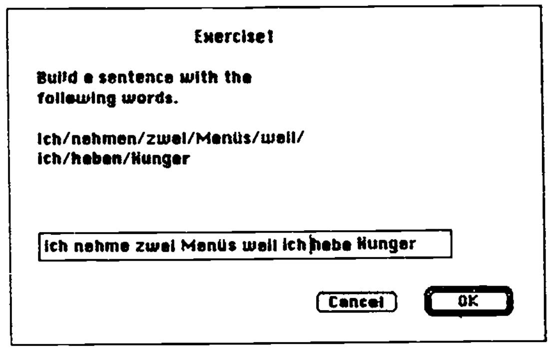


As mentioned previously, as the pool of daemons is enlarged new functionality can be added to an exercise without necessarily perturbing the daemons which have been previously assigned to it. A workbook which includes the exercise in figure 4 may be first produced with the simple set of daemons which we have used in illustration but is not restricted to them. As the program writer becomes familiar with the range of errors which students make in this exercise, the set of daemons can be enriched to accommodate them and added to the exercise.

Building an Intelligent Workbook

The final step in the process of creating a workbook is selecting from the pool of exercises that has been created, those which will appear in the workbook and assigning an order to them. Once this has been done, the workbook is saved as a stand alone application. The workbook displayed to the student consists of a series of windows in which the exercises are presented and the student's input is analysed. An example of a window is given in figure 5.

Figure 5



Conclusion

The work described here is research in progress. We have initially approached the problem of error analysis as one of anticipating possible errors. We believe that this approach has considerable potential, particularly for exercises where students' errors are easily targeted. We have demonstrated that a tutoring system of considerable sophistication can be built from primitive building blocks.

We intend to add further functionality to the lists of actions which a daemon may take and particularly to include natural language processing as part of the error analysis. Natural language processing is particularly good at analyzing grammatical sentences, but has proved difficult to use in error analysis. Parsers normally fail numerous times in the analysis of a sentence and this natural failure is difficult to separate from real errors. Our initial investigations suggest that by assigning natural language processing to a daemon, it is possible to constrain the search space so that the natural language processing system is attempting to solve a particular problem and its success or failure is relevant to the daemon. Our first efforts will include morphological analysis so that students' treatment of inflections can be more deeply analysed. The system described here was written in Allegro CommonLisp™ and runs on the Apple Macintosh™.

References

Alessi, S. M. & Trollip, S. R. (1985). *Computer-Based Instruction: Methods and Development.* Englewood: Prentice-Hall, Inc.

Burghardt, W. (1984). Language Authoring with COMET. *Computers and the Humanities*, 18, 165-172.

Heift, T. (1993). *Error-Adaptive Computer-Assisted Language Learning for German.* M.A. Thesis, Simon Fraser University.

Heift, T. & McFetridge P. (1993) Teaching Subordinate Clauses in German. In *Proceedings of the CCALL2*, University of Victoria, British Columbia, Canada.

Venezky, R. & Osin, L. (1991). *The Intelligent Design of Computer Assisted Instruction.* New York: Longman.

Acknowledgment

This work was supported by a Fellowship from the Advanced Systems Institute of British Columbia.

Interactive Multimedia, Concept Mapping, and Cultural Context

L. HENDERSON, W. PATCHING, AND I. PUTT
School of Education
James Cook University of North Queensland, Townsville., Qld., Australia.. 4811.
E-mail: edldh@jcu.edu.au

Abstract: Concept maps drawn by Aboriginal and Torres Strait Islander tertiary off-campus students were examined to determine the effectiveness of interactive multimedia as an instructional medium for teaching and learning in a multiple cultural context that integrates the requirements of academic culture and aspects of the students' cultures. The quality of the student-generated concept maps was evaluated in terms of hierarchical structure, progressive differentiation, and integrative reconciliation via labelling and directionality of the inter- and intra-level linkages of the relational arcs. Results indicate that concept mapping through interactive multimedia proved an effective meta-learning strategy. Students produced maps at a level of specificity greater than some of those experienced in the actual interactive multimedia courseware.Concept mapping through interactive multimedia is seen as a tool of empowerment in cross cultural learning.

Racial discrimination, enforced dependency, economic exploitation, minimal sustainable political power, educational marginalisation, land alienation and, for some, geographical remoteness, place minorities, such as Australia's indigenous peoples, in the centre of the equity and social justice debate. One program that seeks to address these two areas is the Remote Area Teacher Education Program (RATEP). RATEP, which is an affirmative action initiative driven by the concepts of social justice, culturally-appropriate education, empowerment, and interactive multimedia information technology, provides access to university by enrolling adult Torres Strait Islanders and Aborigines in the Diploma of Teaching at James Cook University of North Queensland through off-campus study in the students' home communities. Diploma subjects are taught by the same lecturers who teach them on campus, the difference being that the subjects are delivered via the computer, other electronic technology (facsimile, audioconferencing and electronic mail), print material, and on-site tutors.

The Issues

Bruner, Goodnow, and Austin (1977) assert that people categorise all objects and aspects of objects to make sense of the complexities of their environment. Categorising involves identifying and sorting stimuli into classes (concepts) on the basis of common criteria. Once formed, concepts or categories "act as intellectual magnets that attract and order related thoughts and experiences" (Martorella, 1986, p.185). Categorisation is, in fact, the principal method of socialisation in any culture, and it reflects the culture in which the concepts have meaning (Howard, 1987). It is therefore not surprising that first year RATEP tertiary students typically find concept attainment within an academic context problematic. In academic genres, they focus on the examples and lose sight of the concept being clarified and, if asked to define a concept, give examples and fail to concentrate on the main concept categorisations and the links between those concepts (McDonald, 1989). Hierarchical progressive differentiation of concepts, particularly with respect to essay construction, also proves troublesome.

Optimising the construction of knowledge in academic contexts by Aborigines and Torres Strait Islanders through interactive multimedia (IMM) courseware requires teaching through, and providing student-generated avenues for, concept mapping in ways that rationally combine two different cultural logics. The first is the specific requirements of mainstream academic culture. This is expressed through the content to be taught and assessed, the written and oral genres, and culturally–specific ways of promoting cognitive development within a tertiary environment. The second, cultural appropriateness for empowerment and ownership, includes the minority's knowledge and methods and conventions of learning in ways that go beyond tokenism.

The paper explores the instructional design of IMM in one RATEP subject, *Australian Minorities Today in World Perspective* (*Minorities*) (Henderson, 1993) to combine rationally the requirements of tertiary academic study with the students' cultural backgrounds and appropriate ways of learning, and to use concept maps as an instructional and learning tool. Specifically, results of fine grained analysis of student-generated concept maps are reported to demonstrate the effectiveness of these interactional teaching strategies.

Welding IMM, Concept Mapping, and Cultural Context

Concept mapping is used as an instructional and learning tool in the subject, *Minorities*. Research by Dempsey (1990) suggests that students who construct their own matrix as a learning strategy retained less information about the concepts than those who were taught via the teacher's matrix. Willerman and Harg (1991) support this in terms of the expert prepared concept map as an advance organiser. Other research shows positive outcomes of student–prepared concept maps (Novak and Gowin, 1984; Moreira, 1985; Sharples, 1991).

As a result, *Minorities* includes both lecturer and student constructed concept maps. Lecturer-prepared concept maps and schemata are important as models and can provide an initial sense of security. For students, the act of map making reveals their degree of understanding and ability to organise, categorise, and progressively differentiate academic content. Student constructions are also a powerful affirming process when working cross–culturally because they allow RATEP students to map concept relationships in a way that may (or may not) reflect a culturally specific view of the domain being studied. To help ensure that RATEP students constructed their own relational maps, a simple ColorPaint shareware program was utilised rather than concept mapping software such as SemNet. Heeren and Kommers (1992) claim that commercial concept mapping software (devised for mainstream students) requires students to use a certain method of knowledge representation. Levesque and Brachman (1987) warn that there is a trade-off between expressiveness, that is, freedom for users to utilise their own mapping style, and tractability in concept mapping software programs, usually to the detriment of expressiveness.

Concept maps are used in a variety of ways in *Minorities*. One is as an advance organiser. Willerman and Harg (1991) point to the positive learning outcomes of utilising concept maps in this way. Initially the concept map is presented in its completed form. Discussion via text and voice over lead students through the map and explains its nature, purpose, and structure as it relates to the particular domain categorised. Then it is rebuilt as it would have been constructed to emphasise one methodological strategy. The content is then restressed to bring the students back to the interrogation of the concept. Such a presentation beginning with the finished product and utilising the visual and aural characteristics of IMM conforms to studies which suggest that Aborigines and Torres Strait Islanders are traditionally oral/aural learners with cognitive strengths in visual and aural information processing and who prefer material to be presented holistically and then broken into parts for particular mastery (Harris, 1984; Kearins, 1982; Osborne, 1982, 1986).

The subject also involves a variation of the standard structure of a concept map. There is a multiplicity of representations which is important in constructivist pedagogy. Such an instance occurs with a map based on the concept that *cultural change*. A major objective was to address the initial concern that students focus on the examples rather than the concepts they clarify. When the concept and its parts were initially presented, the examples faded out leaving only the concepts. Sound was used to effect, too. One sound accompanied the sequential addition of the concepts and a noticeably different softer sound from that accompanying concepts occurred with each set of examples. This strategy exploited visual and aural clues to help RATEP students realise the relative importance of concepts and examples. Academic expository genre is also reinforced. The culminating screen, with the re-inclusion of the examples that had previously been faded-out, is not only a concept map; it also represents an essay plan of the topic: *Culture is not static*. In this way, the concept map is utilised as a psychological structure of knowledge as well as a logical structure of knowledge (essay plan) (Wandersee, 1991).

Another major use of concept maps in the subject, *Minorities*, is as question–answer–feedback interactions that interrogate how well the students understand the concepts presented. They frequently occur as click–and–drag interactions which seem a simplistic tool for learning. Notwithstanding their rudimentary features, click–and–drag interactions utilising IMM characteristics nevertheless provide effective reinforcement of concept attainment because students have to choose between items that are examples (those that stay in the concept box) or non–examples (those that return to their original place for subsequent selection) of the concept.

Even though the exemplars are programmed to belong to a particular concept category, the interaction demonstrates that concepts and concept systems can have unclear boundaries so that some exemplars could belong to more than one concept. For instance, in the interaction on culture change, *books/literature* could belong to the *ideas* as well as the *artefacts* or *colonisation* categories; the *computer* is an example of both a *technological cultural innovation* as well as a *cultural artefact*. In both examples, categorisation depends on how the item is viewed. The culminating interaction requires the students to critique the finished click-and-drag concept map. It provides them with the opportunity to discuss these aspects of "fuzzy category systems" (Howard, 1987) in the exploration of the programmed and their own hypotheses, and also reinforces the notion that concepts and schemata are cultural constructions. Literature on concept teaching sometimes ignores its multipurpose nature.

Concept maps are also used as summaries and, as such, represent a selective synopsis. They are usually designed as interactions with critiques. They occur during the presentation of a topic as well as at the end of the topic. In this way, they highlight effective metacognitive strategies; they call for the students to think about the appropriateness or otherwise of their learning, and synthesising and categorising strategies, as well as to think about their evaluation of their knowledge of a concept as they progress through the topic.

Anatomy of a Concept Map

A concept map is a map of cognitive structures reflecting the psychological structure of knowledge (Wandersee, 1991). It provides a technique of graphically delineating a two dimensional hierarchical–horizontal linear representation of concepts (lexical nodes) and the relational arcs linking those concepts (Norlin, 1980 cited in Harris & Grandgenett, 1993). Relational arcs connecting lexical nodes form propositions indicating the relationship between the superordinate concept and selected salient subordinate lexical nodes within the structure of a particular domain (Bayerbach, 1988). Sometimes a concept map is a schemata for various concepts; other times it is a map of the attributes and exemplars and non-exemplars of a particular concept (Howard, 1987). Concept mapping relates to such Ausubellian principles as prior knowledge, subsumption, progressive differentiation, and integrative reconciliation; it is a metalearning strategy designed to assist students learn how to learn (Wandersee, 1991, p.927) and teachers to teach.

The major components of concept maps that were investigated in the present study were: (i) the hierarchical structure and progressive differentiation of the lexical nodes and (ii) evidence of integrative reconciliation via labelling and directionality of the inter- and intra-level linkages of the relational arcs.

Research Findings

Twenty-one RATEP students were asked to construct on paper their own map on the concept of *culture*. This task was undertaken two months after the students had completed study of the topic on culture and cultural change in *Minorities*. Immediately before drawing their concept map, the students were required to interact with IMM stimulus material which comprised a modified segment from *Minorities* on cultural change. This involved two of the uses of concept maps discussed previously. Firstly, the sequential build-up on screen of a concept map for *cultural change* involved the progressive differentiation of subordinate concepts and examples linked by unlabelled uni-directional relational arcs. Secondly, the students completed a click-and-drag concept map interaction on the same concept.

The students' concept maps of the concept, *culture*, were examined to address three questions:

1) how effective was IMM as an instructional medium for teaching and learning concept mapping;
2) how effective was the IMM courseware in facilitating learning that reflects teaching for cultural contextualisation; that is, the integration of the academic requirements of the subject and aspects of the students' own culture; and
3) to what degree do the concept maps exhibit technical components of hierarchical structure and progressive differentiation of lexical nodes, and integrative reconciliation as exhibited by the directionality, labelling, and inter- and intra-linkages of the relational arcs.

In focussing on the first two questions, cognisance was taken of Reeves' (1992) contention arrived at after a survey of the literature that qualitative methods as opposed to quantitative methods have much to offer understanding of students' learning as a result of exposure to IMM courseware, particularly in light of "the poor record of quantitative research to advance instructional understanding" (p.11).

To answer the first question, the 21 concept maps were examined to see to what extent the they were informed by three things: (i) the actual IMM subject, *Minorities*; (ii) the immediate IMM stimulus screens; or (iii) the students' own everyday prior experiences. In a subjective rating by the authors, the extent of influence of each of the three inputs was rated as either very strong, moderate or minimal. All three components were present in the 21 concept maps. Ten of the concept maps showed very strong evidence of being informed by the subject, *Minorities*, 5 by the stimulus screen and 6 by prior experiences. Six of the concept maps showed moderate evidence of being informed by the subject while 12 and 13 maps by prior concepts and screen respectively. Five of the concept maps were informed to a minimal degree by the subject and three each by screen and prior experiences. Together, 18 or approximately 86% of the students showed very strong or moderate evidence of being informed by the IMM instructional courseware via either the subject or immediate stimulus screens. One could postulate that the different ways of using concept mapping, as advance organisers, variations of concept map structures, click-and-drag activities, and summaries, as components of IMM instructional design do in fact bring about student learning of concept map construction.

With reference to research question two, the concept maps were examined to ascertain the degree of alignment that the lexical nodes had with Aboriginal and Torres Strait Islander culture and general academic cultural requirements. Again, the researchers used subjective evaluation that reflected very strong, moderate or minimal cultural contextuality. All 21 showed that to varying degrees they aligned themselves with both the academic and Aboriginal or Torres Strait Islander cultures. Ten of the students aligned the lexical nodes very strongly with their own Torres Strait Islander or Aboriginal culture, seven moderately, and four minimally while nine of the students aligned their nodes very strongly with the general academic culture, seven moderately and five minimally. There are two points worthy of note. Firstly, 85% of the concept maps reflected Aboriginal and Torres Strait Islander concepts to a strong or moderate degree. This cultural influence is in keeping with research that shows that new concept meanings can only be internalised if they are linked meaningfully with pre-existing concepts (Ausubel, 1963; Gagne, 1985; Novak & Gowin, 1984). Secondly, and more importantly, 80% showed evidence of western academic conceptualisation to a strong or moderate degree. Most students, at least in this exercise, show a significant level of bi-culturality with the successful internalisation and linking of academic and indigenous conceptualisation.

In seeking answers for question three, various technical aspects of the concept map construction were addressed as follows:

Hierarchical Structure and Progressive Differentiation

It needs to be stressed that there can be no one *right* concept map. However, the basic notion underlying hierarchical structure is that learners' cognitive structures are hierarchically organised "with more inclusive, more general concepts and propositions superordinate to less inclusive, more specific concepts and propositions" (Novak & Gowin, 1984, p.77). An examination of the 21 concept maps revealed that all showed evidence of hierarchical structuring of the lexical nodes from more general to specific concepts and examples.

Cultural change was presented as the superordinate node in the IMM stimulus material. In this research exercise, students were asked: "Based on your knowledge, develop a concept map about culture". All of the students used culture as the superordinate lexical node. The authors then counted the number of lexical nodes at the various subordinate levels. Twenty-one (100%) had lexical nodes at subordinate level one, 18 (86%) at level two, 16 (76%) at level three, 10 (48%) at level four, four (19%) at level five and one (5%) at each of levels six, seven and eight. This demonstrates that about one half of the students produced lexical nodes over four subordinate levels. It is interesting to note that at levels one, two and, in most cases, three, most students produced concepts for their lexical nodes. At the lower subordinate levels, some at three, and four through eight, students tended to provide examples of concepts. This is an important finding when viewed in the light of McDonald's (1989) research which found that Aboriginal and Torres Strait Islander tertiary students tend to focus on the example rather than the conceptual level. In this study there is strong evidence to suggest that, as a result of the IMM instructional courseware, students were able to conceptualise at progressive differentiating levels and, then, exemplify where appropriate.

Integrative Reconciliation

According to Starr and Krajcik (1990), integrative reconciliation refers to the degree to which learners perceive relationships among concepts as opposed to compartmentalising or isolating concepts. This is shown on concept maps via the linkages between lexical nodes, both hierarchical across levels (that is, between superordinate and/or levels of subordinate concepts) and across branches (that is, between concepts on the same level). In this study labelling and directionality of the relational arcs were also investigated. Evidence in concept maps of integrative reconciliation is significant in that it can lead to "new and more powerful understandings" by students (Novak & Gowin, 1984, p.104).

(a) Labelling

Labelling of the relational arcs was designated as either explicit or implicit. In the former, students actually labelled the arcs while the latter were unlabelled. Concepts linked by relational arcs form propositions; when the arcs are labelled, they create propositional statements. These propositions, along with arrows indicating relationship directionality help to more precisely develop the connection between linked concepts (Starr & Krajcik, 1990; Fisher, 1990). Explicit labelling was judged to be either simple or complex. Simple labelling was characterised by the verb *to be* (for example, *is* or *are*) or a preposition (for example, *of*) while complex labelling involved the use of clauses, phrases or transitive verbs. An examination of the concept maps reveals that two (10%) of the students used explicit labelling only, 10 (48%) implicit labelling only, and nine (43%) used both explicit and implicit labels. Of those who did label, 11 (53%) did so explicitly. More importantly, of the 11 who labelled explicitly, two (18%) used simple labels, seven (64%) used complex, and two (18%) used both simple and complex labels. That is, 82% of the people who used explicit labels gave complex propositional statements. In total, the concept maps contained 71 complex and 43 instances of explicit labelling, that is, a ratio of approximately 5 complex to every 3 simple labelled relational arcs. Thus, it seems that when students label explicitly, they are more likely to use ones that are complex.

Fisher (1990) argues that the act of labelling or naming relations does not appear to be part of our ordinary, conscious thought processes. We perceive relations but only label them when required to do so for a specific task. In particular domains we develop a shared vocabulary for the concepts but not for describing (labelling) the relational arcs. Thus analysing and labelling relations is "the most difficult step" in constructing concept maps (Fisher, 1990, p.1003). Thus the fact that 10 students used only implicit labels should not be seen too negatively. Nevertheless, the challenging step/act of labelling appears to be an important instructive process as it assists the map maker to clarify relational links between concepts and convey his/her thinking/understanding to the reader. What is significant is the number of students (11 or 53%) who went beyond the IMM concept mapping models to include simple, and most importantly, complex relational naming, particularly as the original and immediate stimulus IMM screens on cultural change did not provide a good model as propositional labelling was implicit.

(b) Directionality

Relational arcs express meaningful relationships, either implicitly or explicitly, between concepts/examples embedded in the concept map. When directionality (either uni- or bi-directionality) is indicated by students through the use of arrows or labelling on the linkages, enhanced clarity of students' understanding of the complexity of these meaningful relationships is expressed. Non-directionality without propositional statements indicates association but not necessarily expressed meaningful relationships.

Of the relational arcs that were implicit, 18 (86%) of the students provided ones that were only non-directional, 13 (65%) only uni-directional, one (5%) only bi-directional, 10 (50%) both non-and uni-directional, and one (5%) whose concept map contained relational arcs that were non-, uni- and bi-directional. Of interest here is the fact that of those students who constructed relational arcs that were implicit, 14 (70%) gave directionality to those arcs. Of the 11 students who provided explicitly labelled relational arcs, one (9%) had concept maps that included only non-directional arcs, 7 (64%) were only uni-directional, none were only bi-directional, one (9%) was both non- and uni-directional, and one (9%) had non-, uni- and bi-directionality. Again, the significance is that 10 (91%) of the students whose concept maps contained relational arcs that were explicitly labelled provided directionality to those arcs thus making clear the students' conceptualisation.

(c) Inter- and intra-level linkages

An examination of the maps showed that by far the majority were inter or cross level linkages with 539 instances of implicit and 114 explicit linkages. There were only 29 instances of intra-level or cross branch linkages. All of these were implicit and provided by only five of the 21 students. This lack of cross branch linkages suggest that the students in this study have not indicated a meaningful understanding of the horizontal relationship between groups of concepts. This is probably a reflection of the IMM concept mapping courseware which did not incorporate elements of cross branch linkages, once again indicating the powerful role modelling plays in the instructional process.

Discussion

Given that all students had not been taught about or learnt through concept maps before studying the *Minorities* subject, the IMM modelling of concept maps has proven successful in a number of ways. There is a variety of student-generated concept maps with no two exactly alike. This is not surprising because, as Novak and Gowin (1984) point out, concept mapping is an idiosyncratic exercise. What is pleasing is that the student-generated concept maps are not replicas of the maps modelled in the IMM courseware or stimulus materials. The variety of concept map types in the IMM courseware would suggest to students that there is no one right concept map and give them confidence to take charge of their own meaning making. This constructivist approach has worked successfully. Students have clearly understood the nature and function of concept maps from the IMM modelling and been able to construct their own representation of the knowledge domain at levels surpassing those presented through the IMM courseware. Their demonstrated ability in concept mapping would now be an advantage when seeking to plan essays on any given topic.

Conclusion

There is little reported work on the instructional design interface between IMM, concept mapping and cultural difference. The proposed paper which combines these three areas adds worthwhile dimensions to the literature. The results from this study indicate that the use of IMM in the subject, *Minorities*, was appropriate as an instructional medium for teaching and learning concept mapping. It was also successful in facilitating learning that reflected teaching for cultural contextualisation. There is additional evidence that the students learned the technical aspects of concept maps, particularly hierarchical structure and progressive differentiation. To a lesser extent integrative reconciliation was displayed even though this was not a feature of the concept maps in the IMM presentations. It would seem that IMM is a powerful tool for the modelling of students' learning concept mapping. Obviously, future IMM instructional design should incorporate aspects that indicate integrative reconciliation, namely complex explicit propositional statements (labelling) and intra-level cross branching.

Reference List

Ausubel, D. P. (1963). *The psychology of meaningful verbal learning*. New York: Gruine & Stratton.

Bruner, J., Goodnow, J., & Austin, G. (1977). *A study of thinking*. New York: John Wiley.

Bayerbach, B. A. (1988). Developing a technical vocabulary on teacher planning: preservice teachers' concept maps. *Teaching and Teacher Education*, 4(4), 337-347.

Dempsey, J. (1990). The concept matrix as a cognitive strategy: an interpretive inquiry into concept training.

Fisher, K. M. (1990). Semantic networking: The new kid on the block. *Journal for Research in Science Teaching*, 27(10), 1001-1018.

Gagne, E. D. (1984). *The cognitive psychology of school learning. Boston*: Little, Brown, & Co.

Harris, S. (1984). Aboriginal learning styles and formal schooling. *The Aboriginal Child at School*, 12 (4): 3–23.

Harris, J. B. & Grandgenett, N. F. (1993). A developmental sequence of children's semantic relatonships: Implications for the design of interactive hypermedia materials. *Journal of Educational Multimedia and Hypermedia*, 2(1), 83-101.

Heeren, E. & Kommers, P. (1992). Flexibility of expressiveness: A critical factor in the design of concept mapping tools for learning. In P. Kommers, D. Jonassen, & J. Mayes (Eds.). *Cognitive tools for learning*. NATO ASI Series F: Computer and Systems Sciences, Vol. 81 (pp. 85-102). Berlin: Springer-Verlag.

Henderson, L. (1993). Australian minorities today in world perspective. 30mb interactive multimedia computer courseware. (2nd ed). Townsville: James Cook University of North Queensland.

Howard, R. (1987). *Concepts and schemata: An introduction*. London: Cassell Education.

Kearins, J. (1982). Cognitive strengths of Australian Aboriginal children. *Wikaru*, 11 (December): 2–19.

Levesque, H. & Brachman, R. (1987). A fundamental trade-off in knowledge representation and reasoning. In R. Brachman & H. Levesque (Eds.). *Readings and knowledge representation*. Los Altos, California: Morgan Kaufmann.

McDonald, H. (1989). Succeeding against the odds: Torres Strait Islanders at university. In N. Loos & G. Miller (Eds.) *Succeeding against the odds: The Townsville Aboriginal and Islander Teacher Education Program* (pp.187-204). Sydney: Allen & Unwin.

Martorella, P. (1986). Teaching concepts. In J. Cooper (Ed.). *Classroom teaching skills*. (3rd edn.). (pp. 181-223). Lexington, Massachusetts: D.C. Heath and Company.

Moreira, M. (1985). Concept mapping: An alternative strategy for evaluation. *Assessment and Evaluation in Higher Education*, 10 (2): 159–168.

Novak, J. & Gowin, D. (1984). *Learning how to learn*.. New York: Cambridge University Press.

Osborne, A. (1982). Field dependence/independence of Torres Strait Islander and Aboriginal pupils. *Journal of Intercultural Studies*, 3 (3): 5–18.

Sharples, M. (1991). Computer–based tutoring of visual concepts: from novice to experts. *Journal of Computer Assisted Learning*, 7 (2): 123–132.

Starr, M. L. & Krajcik, J. S. (1990). Concept maps as a heuristic for science curriculum development: Towards improvement in process and product. *Journal for Research in Science Teaching*, 27(10), 987-1000.

Willerman, M. & Harg, R. (1991). The concept map as an advance organizer. *Journal of Research in Science Teaching*, 28 (8), 705-711.

Strategies for Integrating Computer-Based Training in College Music Theory Courses

George J. Hess, Jr.
Music Technology Center, School of Music,
University of Northern Colorado, Greeley CO 80639
e-mail: georgehjr@aol.com

Abstract: During the fall semester of 1993, a curriculum-based CBT program was used to replace all in-class drill in intervals and chord identification for one section of freshman music theory at the University of Northern Colorado. A comparison with a section using commercial software as an optional supplement to traditional in-class drill found no significant difference in achievement or attitude. However, the results indicate the computer provided more efficient and consistent instruction. There was also a significant difference in the amount of time spent outside of class on ear training and in the amount of improvement attributed to the use of CBT.

It is difficult to imagine a prerequisite for serious music study more important than the ability to hear and identify musical sounds accurately. Indeed, virtually all music schools incorporate some sort of aural training in their lower divisior curriculum, most often in the form of sight-singing and dictation classes (Pembrook & Riggins, 1990). The problem is that, due to disparate abilities and rates of learning, peer-pressure, delayed and ineffective feedback and the lack of class time available, aural skills are most efficiently and effectively acquired through private, individualized instruction.

It was proposed that computer-assisted instruction had the potential for solving the needs of many students. In the twenty-five years since research in computer-assisted ear training instruction began, numerous studies have compared computer-based instruction with traditional in-class drill or pre-recorded programmed instruction. Among the reported benefits were (a) better performance on ear training tests, (b) better long-term retention of aural discrimination skills, and (c) improved attitude towards ear training and computer-assisted instruction. These benefits were primarily attributed to the computer's ability to provide individualized instruction and immediate evaluation and feedback. (Allvin, 1971; Benward, 1987; Hofstetter, 1981).

Purpose

Computer-assisted ear training instruction has been shown to be effective in numerous studies during the 1970's and 1980's (Hofstetter, 1975; Humphries, 1980). In many of those studies, courseware was specifically developed to support their respective curricula and its use required. In contrast, since the development of the microcomputer, most music theory courses have relied on the voluntary use of generalized, commercial software to provide computer-assisted drill (Pembrook, 1990). The intent of this study was to determine if a required, curriculum-specific, computer-based ear training program could be used as an effective substitute for in-class drill, individual practice and the optional use of commercial software. Additionally, it was expected that information would be obtained regarding the effectiveness of the courseware used in the study. A secondary purpose was to determine whether attitudes toward ear training and computers were affected by the different methods.

In the course of the study the specific questions that were to be investigated are: How does each method affect student progress and performance? How does each method affect the amount of time spent outside of class practicing ear training and the effect on the student perception thereof? How does each method affect students attitudes towards ear training and computer-assisted instruction? What are the effects on students attitudes when

CAI is required as opposed to voluntary? What is the relationship between scores on practice drills and examinations? What additional benefits and drawbacks are perceived by the students in regards to each method?

CAI Program Design

A computer-based learning system was developed by the researcher to correspond with the University of Northern Colorado music theory curriculum. Based on discussions with the music theory faculty, five subject areas were selected to be included included in the software--Intervals; Tone Rows; Soprano, Chord Type & Bass; Chord Qualities; and Diatonic Progressions--with each subject being presented in three ways. The *Lessons* present concepts and identification strategies from as many viewpoints as possible. Students are encouraged to interact with the program in a variety of ways and the lessons provide stimulus in the three modes of learning: auditory, visual and kinetic. Interspersed throughout the tutorials are open-ended, ungraded, guided *Practice Sessions*, which are also accessible directly from a submenu. The *Drills* have from three to five levels with preset contents and an option to customize the exercises. Upon completion of a drill, the percentage of correct answers is calculated and displayed with a suggested path for further study.

The program stores thorough records regarding student activity and progress. Students must be registered to use the program. When students first complete the registration process, a new personal records file is created on the file server. Upon the completion of each practice exercise or drill, the program opens the student's record file and records information about the current session. Depending on the type of instruction, the stored information may include the name of the lesson, the time spent in the section, the number of questions, the percentage of correct responses and the number of replays per exercise. Upon electing to exit the program, a summary of the total number of exercises and the average score is shown.

Navigation is accomplished through a series of menus and submenus. Structure within the tutorial and exercise sections is primarily linear, with some optional branching for additional information. The structure of the drills is more independent, with each drill accessible only from its respective submenu. Students may exit at any time. The sound is MIDI-based and students are encouraged to select a sound with which they are comfortable.

While research indicates that total learner control results in the least improvement (Steinberg, 1977), it was also noted that the control maintained by competency-based programs increased student frustration (Hofstetter, 1975). The compromise chosen, based on Tennyson's model (1981), was to allow learner control with advisement. Students may select any section of the program at any time; there are no prerequisites. The program provides advice as to the suggested course of study and also influences decisions by limiting choices and through screen design.

Procedures

The study was conducted during a ten-week period of the Fall 1993 semester in the University of Northern Colorado School of Music. One section of the freshman music theory class, "MUS 101-Sight Singing and Music Theory" was selected as the experimental group, with another functioning as the control group. All freshman music theory sections follow approximately the same syllabus and administer the same exams.

Students in the experimental group received all instruction and drill related to harmonic dictation and intervals through the use of the computer-assisted instruction programs developed for this class. The students were instructed in the use of the software during one regular class session. During weeks six through fifteen, students in this group were required to spend a minimum of two hours per week on the programs, unless they earned three scores of 90% or above on the highest level for each subject. The use of the program was averaged as part of the homework and quiz grade which accounts for 20% of the final grade. Students using the program an average of 2 hours per week or successfully completing all levels received a grade of 100; each minute below 120 per week resulted in a one point reduction. Quizzes and examinations took place as usual and included dictation examples.

The control group received traditional instruction, which consists of classroom drill and practice outside of class. Voluntary use of commercial, computer-assisted ear training software was encouraged, however, access to the curriculum-based software was restricted.

Because random selection of the subjects was not possible, a quasi-experimental pretest-posttest, control group design using intact groups was chosen. Of the five existing sections of freshman music theory, one was selected as the experimental group and a second as the control group. All students take a departmental placement test on the second class meeting; this was used as the pretest. The final ear training examination for the course was used as the posttest. The design is represented in Figure 1, with 0_1 to 0_4 representing the pretest and posttest scores, T representing the regular classroom instruction with voluntary use of commercial computer-assisted instruction programs and TX representing the regular classroom instruction with curriculum-based computer-assisted instruction replacing in-class drill.

Group	n	Pretest	Treatment	Posttest
I	23	0_1	T	0_2
II	19	0_3	TX	0_4

Figure 1. Experimental design

The treatment was administered in the University of Northern Colorado Music Technology Center. Students were allowed to choose the time and length of sessions. Availability and convenience was not a consideration as the Computer/Synthesis Lab is available seventy-five hours per week. The lab consists of 31 workstations with Macintosh SE/30 computers and Kawai K-4 or K-11 synthesizers.

A Likert-type, attitude instrument, developed by the researcher was administered to both groups before and after the treatment period. The purpose was to determine if the treatment had any effect upon the attitudes of the subjects toward ear training or computers, and to see how effort was affected.

A test was also administered to determine if any of the subjects possessed perfect pitch. Students were asked to identify eight pitches played on the piano by the researcher. As a result, two students who were found to have perfect pitch in the control group were excluded from the study.

An audit trail was maintained by the computer-assisted ear training software used by the experimental group. Information regarding student progress, time per exercise, total time per session, exercises attempted, number of questions attempted and number of replays required was recorded.

Limitations

Experimental mortality occurred in both groups with three students in each group dropping the class and not taking the final exam, leaving 16 students in the experimental group and 20 students in the control group.

Intrasession history may have affected the results. On a number of occasions, students in the experimental group reported malfunctions in the program. In general, these errors had to do with the storing of records and were not sufficient to affect aural skills development, but may have had an effect upon students' attitudes.

While the synthesizers used have a fair representation of an acoustic piano, the timbral difference between the synthesizer and the piano may also have had an effect on exam performance. Students in the experimental group practiced using timbres found on the synthesizer, but the exam was performed on a piano.

Data Analysis

The study was conducted to determine whether aural skills can be taught as effectively through the exclusive use of a curriculum-oriented, computer-based tutorial and drill program, as when taught by traditional in-class dictation drills coupled with the optional use of commercial software, and to determine the effects of such a method upon the attitudes of music students. A control and an experimental group each received the departmental placement exam and final exam for ear training, which functioned as pretest and posttest, respectively. The two groups received identical tests.

To determine which method provided the greatest improvement in aural discrimination skills, an analysis of covariance (ANCOVA) was performed to equate the groups and compare performance statistically, with the pretest as the covariate and the posttest as dependent variable. Table 1 shows that scores of both groups final ear training examinations were statistically equivalent. While no difference was found between the adjusted means, a decrease of over 30% from pretest to posttest was noted in the variance and standard deviation for the experimental group. At the same time, those measures increased 10% for the control group.

Table 1
Analysis of Covariance

Source	Sum-of-the-Squares	DF	Mean-Square	*F*-Ratio	*p*
Group	0.026	1	0.026	0.000	0.989
Pretest	812.012	1	812.012	5.650	0.023
Error	4742.876	33	143.724		

A comparison of the overall final grades of both groups using a *t* test was made to determine whether the additional time in class made available to the experimental group due to the treatment had an effect. The experimental group's mean final grade was slightly higher than the control group's, but the difference was not significant.

Data was sought regarding the attitudes of the students and the effect the treatment had on those attitudes regarding ear training, computer-assisted instruction, self perception of aural discrimination skills ability ,and expected success Seven questions were each asked of both groups before and after the treatment. The questions are listed in Figure 2.

1. How would you describe your attitude towards ear training study?
2. How would you rate your abilities in ear training?
3. How important (relevant) is ear training to you?
4. What is your attitude towards computers in general?
5. Please rate your knowledge of computers.
6. How do you feel about using computers to replace some in-class instruction?
7. How well do you expect to do in the ear training part of this class?

Figure 2. Attitude Instrument Questions

Students were asked to rate their answers on a five-point Likert scale. A *t* test was performed for each question on the pre-treatment survey. A significant difference at the 1% level was found between the two groups for question 5 -- prior computer knowledge. No significant differences were found on any of the other questions. The *t* test performed on the post-treatment survey indicated no significant differences on any of the questions.

The two questions relating to effort were worded differently on the surveys and are listed in figure 3. On the pretreatment survey the means for both groups were nearly identical with no significant difference. Both questions indicated the experimental group felt they spent significantly more time and effort on aural skills than the control group. The results for both questions are listed below in Table 2.

Pre Prior to this class, how much have you worked on developing your ear ?
Post How much have you worked on developing your ear this semester?

Pre How often do you expect to practice ear training outside of class this semester?
Post How often did you practice ear training outside of class this semester?

Figure 3. Effort questions: pre-treatment and post-treatment surveys

Table 2
t Test between Post-treatment time and effort questions

	Mean Diff	SD Diff	T	DF	*p*
Question 8	0.667	0.840	3.367	17	0.004*
Question 9	0.735	1.200	2.525	16	0.022**

* indicates significance at 1% level ** indicates significance at 5% level

One final question was administered to both groups on the post-treatment survey only: How much of your improvement do you attribute to using the computer? A *t* test comparing the two groups found that the experimental group attributed significantly more of their improvement to the use of the computer. The results are shown in Table 3 and indicate significance at the 1% level.

Table 3
t test on computer effect

	Mean Diff	SD Diff	T	DF	*p*
Question 10	1.176	1.639	2.960	16	0.009*

* indicates significance at 1% level

A number of correlations were performed using data from the posttest and audit trail as coefficients to determine whether a relationship exists between progress and mode of instruction, time spent per section, time spent per session, number of sessions and number of replays per exercise. No significant correlation could be found between these aspects.

A correlation was performed between the number of levels attempted in the program and the final ear training examination score. Table 4 shows this relationship to be significant at the 5% level.

Table 4
Correlation of Exam Score and Levels Attempted

	Final Exam	Levels	
Final Exam	1.000		
Levels	0.580	1.000	
Bartlett Chi-Square Statistic:	4.314	DF= 1	PROB= 0.038

Comparisons of the students scores in the program drills and the posttest scores were used to determine how strong the link was between computer-assisted instruction program and the music theory curriculum and to determine if progress in the computer program could be used to predict success in the music theory examinations. The correlation shown in Table 5 shows the relationship to be significant at the 5 % level.

Table 5
Correlation of Exam Score and Average Drill Score

	Final Exam	Avg Score	
Final Exam	1.000		
Avg, Score	0.617	1.000	
Bartlett Chi-Square Statistic:	5.031	DF= 1	PROB= 0.025

Conclusions

The following conclusions are based on the results of the study. Any generalizations are limited to the freshman theory program at the University of Northern Colorado and the software used in the study.

The results indicate that curriculum-specific, computer-based training can be used as the primary method for the acquisition of aural skills. Students using the computer-assisted instruction exclusively developed slightly higher levels of aural skills than did students who used traditional methods, however the difference was insignificant. The grade distribution of the final exam indicated 50% of the experimental group received a B or better (37.5% received an A), compared with 40% of the control group (15% received an A).

No significant differences were found when comparing the overall final grades of the two groups, nor was there any difference in the gap between the final grades and the ear training final. As such, there is no indication that any benefits were derived from the additional time made available to the experimental group by the exclusion of in-class dictation drill. Possible reasons for this are that the study did not control for instructor effect and that the written skills of the students were not statistically equated. Half of the experimental group was repeating the class, identifying them as possible low-achievers. Nonetheless, it ishould be noted that the instructor of the experimental group felt the method allowed him the flexibility to spend extra time on topics when necessary.

A primary benefit of the use of curriculum-based aural skills software is its efficiency and individualization of instruction. The decrease in the variance from pretest to posttest for the experimental group indicates the computer program provided more consistent instruction. This is all the more interesting considering the number of remedial students included in the group. It also appears to confirm the effectiveness of providing learner control with advisement. Students were allowed to choose the content, level and type of instruction and it would appear that most students made wise choices.

No change in attitude towards ear training resulted from the use of the software. Student attitudes remained positive in both groups; possibly a reflection of the overall success achieved. Both groups were also convinced of the relevance of aural skills training. Attitudes towards the use of computer-assisted instruction remained constant as well, even though students in the experimental group believed the software was effective and that its use was responsible for most of their improvement, while students in the control group felt their use of commercial software had little effect on their abilities. The experimental group was also more optimistic about their chances on the final examination, even though the control group was slightly more accurate in their predictions.

Keller suggested four factors--maintenance of attention, relevance of the material, student confidence and student satisfaction--are required to provide motivation (cited in Alessi, 1991). Coupled with the course requirement to use the program, this appears to be have been effective. Students in both music theory classes appeared to be reasonably motivated students, and indicated they expected to spend approximately the same amount of time outside of class working on aural skills. However, students in the exper`mental group felt they spent significantly more time outside of class than the control group and averaged just over 80 minutes per week using the program.

References

Alessi, Stephen M. & Trollip, Stanley R. (1992). *Computer-Based Instruction: Methods and Development.* Englewood Cliffs NJ: Prentice Hall.

Allvin, Raynold L. (1971). Computer-assisted music instruction: A look at the potential. *Journal of Research in Music Education, 19,* 131-43.

Benward, Bruce (1987). *Ear Training: A Technique for Listening (3rd ed).* Dubuque IA: Wm. C. Brown.

Hess, Jr., George J.(1994) *Dictation Tutor: the Effectiveness of a Curriculum-specific Tutorial in the Acquisition of Aural discrimination Skills for College Music Students.* Doctoral dissertation, University of Northern Colorado, 1994.

Hofstetter, Fred T. (1981). Applications of the GUIDO system to aural skills research, 1975-80. *College Music Symposium, 21* (2), 46-53.

Hofstetter, Fred T.. (1975). GUIDO: An interactive computer-based system for improvement of instruction and research in ear training. *Journal of Computer-Based Instruction, 1* (4), 100-106.

Humphries, James A. (1980). The effects of computer-assisted aural drill time on achievement in musical interval identification. *Journal of Computer-Based Instruction, 6* (3), 91-98.

Pembrook, Randall G. & Riggins H.Lee (1990). "Send Help!": Aural skills instruction in U.S. colleges and universities. *Journal of Music Theory Pedagogy, 4* (2), 231-242.

Steinberg, E. R. (1977). Review of student control in computer-assisted instruction. *Journal of Computer-Based Instruction, 3,* 84-90.

Tennyson, Robert D. (1981). The use of adaptive information for advisement in learning concepts and rules using computer-assisted instruction. *American Educational Research Journal, 18,* 425-38.

COMPANION : An interactive learning environment based on the cognitive apprenticeship paradigm for design engineers using numerical simulations

Y. HILEM & M. FUTTERSACK
LIUM
Université du Maine, Route de Laval, BP 535, 72017 Le Mans, France
E-Mail : youcef@lium.univ-lemans.fr

Abstract: The use of numerical simulations to design and analyze new industrial products requires both conceptual and operational knowledge. This paper describes COMPANION: an intelligent multimedia system in vocational training. The system is used by engineers and technicians on-line during their work in a company. A key concept in terms of training methodology is that of "situated learning" or continuous learning in the workplace.

1. Introduction

During the design of new products, some engineers are involved in computer simulations to analyze the behaviour of the product with complex and specialized tools called Finite Element Simulation Softwares (FESS). As all complex computer tools, the traditional help systems in the form of on-line manuals and interactive tutorials are not efficient to master the corresponding tools [Larsson, J.E., Persson, P. 1988; Matthews, M.M., et al. 1988; Sandberg, J., et al. 1988]. In addition, in educational context a computer-based simulation should have additional knowledge (conceptual) of the domain in order to provide the user with a more deep understanding of the underlying model [Berkum, J.V., et al. 1989]. Finally, a FESS is a Computer Aided Engineering (CAE) tool where difficult tasks such as modelling, solving and analyzing engineering problems are performed. Thus, the use of a FESS is not simply limited to the knowledge of the commands but requires also the knowledge of the problem solving methods. On the other hand, in industrial (vocational) training contexts, the main objective is improving qualification and proficiency, i.e. acquire operational knowledge (skills) through repeated experiences of related tasks. To achieve all these features, computer-based environments for FESS should integrate both conceptual and operational knowledge acquisitions. In our research, a key concept in term of vocational training methodology is that of "situated learning" or continuous learning on the workplace, i.e. in the culture and context of actually performing the task. A design model supporting situated learning is cognitive apprenticeship which is an extension of the traditional apprenticeship model focusing in learning problem solving skills in the practice of performing authentic tasks [Collins, A., Brown, J. S., and Newman, S. 1989]. The novice (apprentice) watches the expert performing the task and the expert allows the novice to ask questions and perform small parts of the task with the amount of the task carried out by the novice increasing as experience is gained. Instructional principles are: modelling, coaching, scaffolding, fading, articulation and reflexion. This approach seems the most promising for teaching/learning operational knowledge [Cooper Eric, B. W., 1991; Tong, T. 1993; Newman, D. 1989]. For conceptual knowledge, the use of hypertext and hypermedia techniques [Duffy, T.M., Knuth, R.A. 1990; Honebein, P. C., et al. 1992] reveals efficient for adults which want more free browsing among knowledge items.

Section 2 describes the different activities of the student in COMPANION and shows how cognitive apprenticeship and hypermedia techniques are integrated in the whole system. Section 3 focuses in the structure of COMPANION. Section 4 details the content of hypermedia modules used in conceptual knowledge acquisitions. Section 5 describes how the operational knowledge is structured and acquired. Section 6 and Section 7 details, respectively, the engineering activity and the technical assistant which embodies the knowledge that become operational during problem (small projects) solving with collaboration between the student and COMPANION. This paper ends with a conclusion and feature research.

2. COMPANION overview

The COMPANION system is primarily aimed at technical staff (technicians and engineers) using FESS in structural dynamics applied to the fields of stamping (sheet metal forming) and occupant safety. Stamping simulation addresses large manufacturing companies as well as their smaller sized subcontractors. Consequently, the technical staff being trained may have little academic background in mechanics and little or no knowledge in CAD (Computer Aided Design)/CAE(Computer Aided Engineering), but in general plenty of experience in their own field of technology. For occupant safety, the user community is mainly in the automobile industry, and is more acquainted with CAD/CAE concepts.

COMPANION distinguishes three leveis of students:
Base student. He/she has an acceptable level of mechanical engineering and little experience of the finite element method (or none at all). The system is not intended to teach him/her the finite element method. It teaches him/her practical rules of use of the FESS[1]. As a result, the student will be able to perform robust calculations of acceptable quality, mostly elementary calculations and parametric studies.
Intermediate student. He/she has a good experience of standard finite element tools and knows the implicit integration scheme. The system teaches the explicit integration scheme and the particular use of it in the FESS. As result, the student will be able to perform industrial calculations.
Advanced student. He/she has already used non-linear dynamic tools which use the explicit integration scheme. The system teaches the advanced methods of the profession. As a result, the student will completely overcome the FESS.

With a set of questions, the system evaluates the initial level of the student and selects a set of hypermedia modules corresponding to a certain level of use of the FESS (e.g. performing parametric studies). The student can then explore at his/her own pace the different modules which provide a conceptual view of the objects in the domain that are important to performing the tasks, and also their relationships to one another. For example, the contact modelling module (see figure 1) introduces the basic concepts (physical and numerical) required for running FESS. Once the student has explored this module, the system assumes that he/she can use the contact function in the FESS. The student can insert comments and record sound in the hypermedia modules to inform the system that he/she does not understand a concept, he/she needs an example, more detail and an animation. Thus, no evaluation is done to insure that the student is acquiring the knowledge represented in the modules in a way that will be useful to performing the tasks in which he/she is being trained. But, the system checks after each module to see if there is a message left by the student. If there is a message, the system generates a file message to the author of the module. While the student is progressing through the modules, a list of available exercises and problems is updated which are all relevant to the knowledge already acquired. Exercrises (see figure 2) are elementary applications on parameters influence where the system allows the student to choose between values and obtain from the system interpretations and evaluations of the simulation results and comments about the choices.

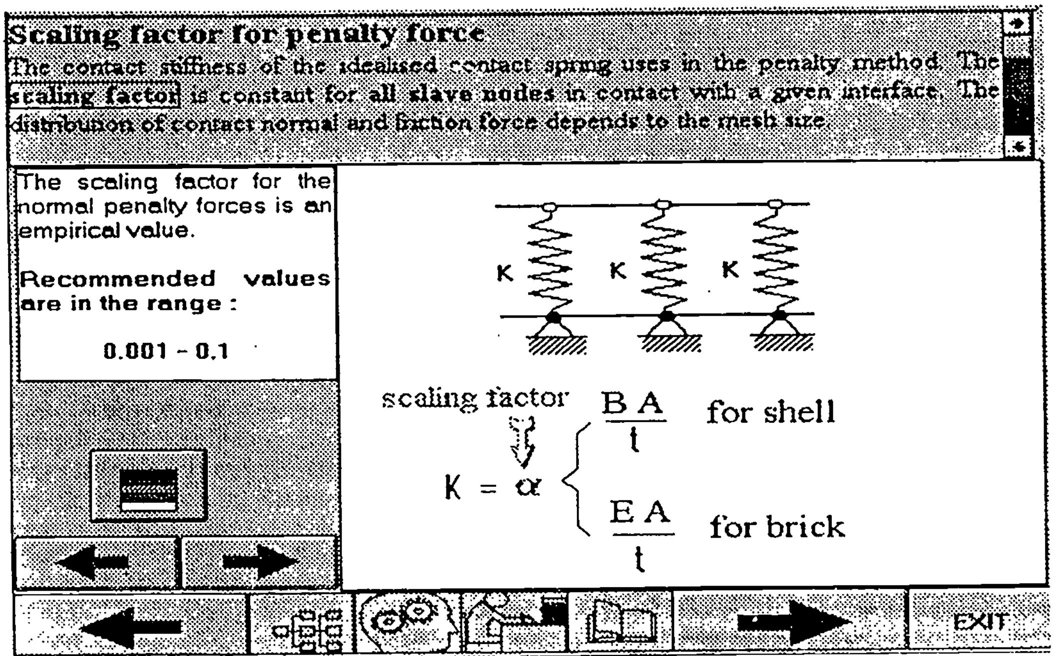


Figure 1: Example of node in contact modelling module

[1]We use the software PAM-SAFE™ for safe and PAM-STAMP™ for stamping.

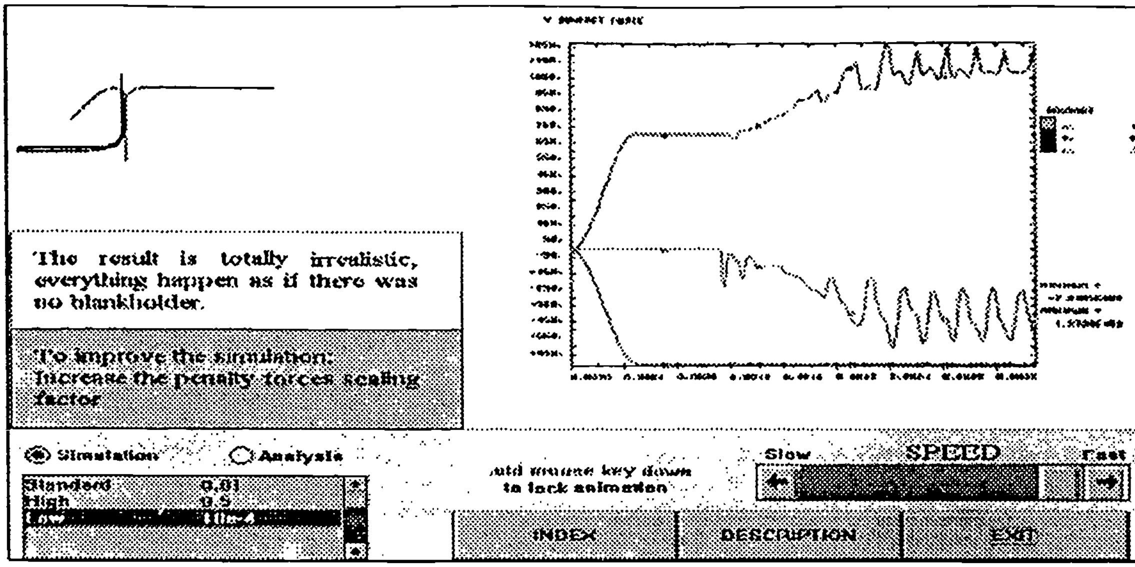


Figure 2. Exercise example

Problems are small projects that cover all the concepts introduced in the modules explored by the student. When the student activates the problem solving, the system execute the two following steps : Standard case - the objective is to focus the student on the knowledge of the use of tools. The system guides the student from general knowledge (sequences of tasks and subtasks) to specific knowledge (sequences of commands). Sensitivity analysis - the main objective is to acquire experience in the analyze of a product by altering parameter values. When the student changes the values, the system provides interpretations of the results (see figure 1 - explanation window) and comments (e.g. "You have increased material density and obtained CPU gain", "Homogenenous strain in specimen - the solution is acceptable", "Strain localized in head specimen - the solution is not acceptable") according to the success criteria associated with the problem. When the student is asked to change a value in order to reach a certain goal (e.g. increase stability, increase cpu gain, reduce the blankholder displacement) the system gives hints (e.g. "When no pressure is applied, material model 100 leads to a significant reduction of running and the storage size", "The iterative method, model 103, leads to a more accurate solution for quadrilateral shells with isotropic yield criterion").

3. COMPANION architecture

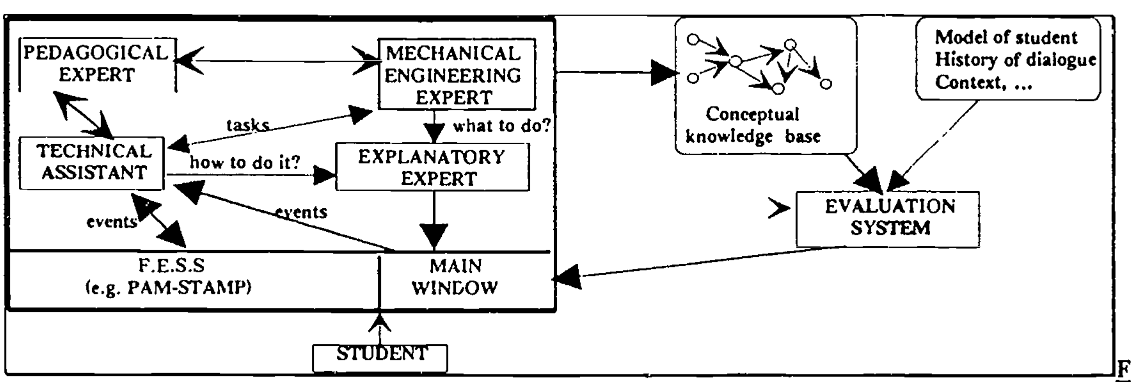


Figure 3. Companion architecture

COMPANION (see figure 3) is a multi-agent system [Futtersack, M., Labat, J.M., 1992] that comprises knowledge-based systems and hypermedia modules: A conceptual knowledge base that contains a detailed description of the modelled domain (network of hypermedia modules); An evaluation system that provides the right hypermedia modules corresponding to the student's level; A pedagogical expert which embodies the model of cognitive apprenticeship; A mechanical engineering expert that contains general engineering knowledge: tasks (e.g. modelling the problem domain), sub-tasks (e.g. adapting the mesh to the objectives of the analysis) and production rules for terminal decisions (e.g. choice of time increment); A technical assistant expert that contains specific knowledge, i.e. functional description of tool interface in term of commands; An explanatory expert that responds to specific requests of the student and provides coherent integrated explanations of both conceptual and operational knowledge. It can also activate a node in a module to provide more details or an animation. COMPANION uses authentic FESS. Thus another functionalities (not visible to the student) are added to the interfaces used in the FESS in order to satisfy the requirements of COMPANION: to be able to proceed the problem solving and to be informed of what is happening in the interface. Each interface is a task (in a multi-task environment) which communicate by messages with the technical assistant.

4. The conceptual knowledge base

The conceptual knowledge base (CKB) contains a detailed description of the modelled domain, the basic assumptions and hypotheses underlying the model (e.g. kinematic hypothesis, contact algorithm), the definitions of concepts, and the meaning of parameters and variable values (e.g. a description of critical values or typical values of variables). This CKB is organized as a hierarchy of hypermedia modules which can be performed independently. Each module has prerequisite relationships and/or isa specialization relationships to the other modules. In a module several concepts are introduced. They are grouped under four categories:

Relevant physical concepts. These part of the modules provide a good physical understanding of the concerned phenomena and an overview on the theoretical considerations needed for the proper resolution of the industrial problems of the student.
Modelling. The modelling aspects are discussed in order to provide a good understanding of the input parameters and the ability to realise reasonable models of the analysed phenomena.
Practical considerations. The specificity of each type of analysis (here stamping or occupant safety simulations) allows the student to access for hints about classical pitfalls.
Exercises. Made accessible by the system when the student has been studying the relevant parts of the module. These exercises are elementary applications (variable assignment exercises) that provide a more understanding of concepts and principles underlying the model. The student is free to choose values to assign to certain variables or parameters and watch, in real time, the development of the state of the model. The system evaluates the given answer to give some comments and interpretations of the results.

Much of phenomena (e.g. self-contact, airbag and stamping) are illustrated by synchronizing animation and sound. Hotwords are also used to trigger pop-up explanations of terms in a text or branch to other nodes.

5. Pedagogical expert

A FESS provides an infrastructure for accomplishing engineer's tasks, i.e. the various activities appear as objects in the screen like the names of menus, dialogue boxes, fields to type in, or buttons. The interface of FESS is generaly an event-driven system where nothing happens until an event occurs. Events result from user actions such as clicking a mouse button, pressing a key, moving the mouse on the screen, or choosing a menu item. We define the student interaction with the FESS as a sequence of operations that are performed upon the interface to do a task. When there is a normative sequence of input and output associated with the interaction we will speak of procedure or skill. The distinction between a procedure and a skill is that the student can have knowledge about procedures without being particularly 'skilled' in executing them. To take into account of some sources of cognitive difficulties [Waern, K. G, 1991] in the interaction with the FESS, the pedagogical expert uses two attributes: ***Task complexity:*** estimates the effort that is required to learn how to master the sequence of operations that compose the task. This attribute takes three values: very complex, complex and simple. The basic idea is that integrating command sequences as procedural knowledge (more or less automatic) for the most frequent work tasks is done quite quickly. ***Memory-problems:*** depending on the number of zooming operations and the time response of commands that are in the task, the pedagogical expert affects to this attribute one of these values: much, few and no. The basic idea is that tasks using the zooming function to enlarge mesh details present some short term memory problems, in that the student may lose the perception of the totality of the mesh. In addition, the computer takes in several cases long time to respond to certain commands and could also introduce short term memory problems. The typical situation in such cases is that the student is conceptualizing a sequence of procedural operations.

The operational knowledge is organized into general knowledge (mechanical engineering expert) which determines what to do and specific knowledge (technical assistant) which determines how to do it. Both general and specific knowledge are acquired in the context of solving a library of problems with collaboration between the student and the system. These problems are grouped according to the level of use of the FESS. The student go over this library from base to advanced level of use. The student has the overall control on problem solving. He/she decides whether to interrupt the resolution and proceed or whether to stop and ask the system to proceed. The system interventions on student failures are immediate in coaching stage and are wait-and-see [Galdes , D. K., et al.,] in fading stage. The system evaluates student actions in terms of the expert understanding of the task [Newman, D. 1989].

Depending on the student's model, the objectives of the problem, the level of use of the FESS, the focus knowledge (use of the tool, comprehension of the product physics, compehension of the task of an engineer, improve the performance of a certain skill), the pedagogical expert selects one or more different intervention types during the resolution of a problem. The intervention types are: to accompagny (e.g. "Lets choose the units"); to direct (e.g. "Introduce the material properties"); to take care by forbid some uses (e.g. "Do not use physical contraints for the tools") or by prevent for the eventuel errors (e.g. "Be sure that the sheet mesh is regular"); to reduce the difficulty (e.g. breaking a goal into its subgoals), summarize, demonstrate a goal; to maintain the orientation in comparison with the principal goal or to the intermediate goals (e.g. to remind the goal); to indicate the decisive characterestics (e.g. provide the useful information to proceed); control the frustration (e.g. provide evaluations, encouragements); to scaffold, i.e., provide a set of mechanisms that enable the student to perform a task, but which fade away as the student becomes more expert (e.g. disable all menu items in the interface that are not useful to perform the task in foccus, or highlight a menu item). This is what called [James A., et al., 1988] dynamic support which is derived from the learning principle referred to as "the zone of proximal development" [Vygotsky, 1986]. The pedagogical expert is a knowledge-based system where each intervention type is implemented as a set of production rules. Another rules maintains the student model (overlay model) after each problem solving.

6. Mechanical Engineering Expert (MEE)

The MEE is a knowledge-based system that contains engineering expertise. It contains a knowledge base structured into two levels: strategic level and a tactical level. The strategic level is described by a set of plans that organize the overall control of problem solving. The tactical level is a set of production rules that make different terminal decisions like the choice of material type, the choice of variable/parameter values. The strategic level is tool independent but the tactical level is tool dependent as shown in the following examples:

Plan 1 **Objective:** resolve a problem Actions: (preprocessing, resolution, postprocessing)[2]	**Rule 5** If the problem is to improve the stability of the computation Then define two bead interfaces, one against each surface of the sheet.
Plan 4 **Objective:** introduction of complementary data **Actions:** {Nodal restraints, Load definitions, Slide interface, Material properties, Control data}	**Rule 11** To introduce an equivalent drawbead between the sheet, the blankholder and the die Use the model type 4.

Table 1: Plans and rules examples

The MEE contains also a plan-interpreter KEPLER-OBJECT which is an extension of KEPLER [Futtersack, M., Labat, J.M. 1993; Vivet, M. 1988] to allow collaborative solving and object-oriented features.

7. Technical Assistant

The Technical Assistant monitors student's interactions with the FESS, makes the useful suggestions to the student about FESS capabilities in the context of current activities and facilitates the accomplishment of a particular task by a student who does not currently know how to do it. It contains a network describing the FESS interface. At the meta level, the nodes represent tools and the links represent events that define the action to perform on an object in order to access to another tool. A tool-node contains a hierarchical organization of actions as they appear in the interface of the FESS. The lowest level correspond to elementary actions (e.g., choice of material model, introduce parameter value) whereas higher level nodes correspond to abstract tasks (e.g., creating the mesh, introducing complementary data, analyzing results) that are executed as a sequence of sub-tasks. (A sub-task could be either a simple action or another sequence of sub-tasks). A sequence of events is associated for each action in order to know how to perform it. The choice of what prompt to give and/or how to do a task are implemented by searching a path between the current state and the goal state. A current state is defined by a set of all visible objects and the goal state by a set of objects that will be visible in order to perform the desired task. The path contains the sequence of events that allow the transition between these two states.

[2] ordered actions are between () and independent actions are between { }.

Depending on the intervention type selected by the pedagogical expert, this exhaustive path is pruned before submitted to the explanatory expert.

8. Conclusion

A prototype COMPANION has been implemented on a PC/Windows platform using TOOLBOOK for multimedia aspects, KAPPA for rule based and oriented-object techniques and KEPLER for the planning. This prototype is tested by some end users and evaluated by trained specialists. The results are globally positive especially for the global philosophy of COMPANION: hypermedia techniques for conceptual knowledge and cognitive apprenticeship for operational knowledge. The organization and the structure of the modules are appreciated and steel the same for the end product. Actually, we improve the expertise of the different knowledge-base systems and add another functonality to the system, where the student can express his/her hypothesis, interpretations and evaluation of the simulation results. Further experimental studies with COMPANION are also planned.

Acknowledgement: The work presented here was carried out under the COMETT 2 project. We gratefully acknowledge the contributions of the following colleagues from the project: E. Di Pasquale (ESI*), P. Culiere (ESI), L. Penazzi (ESI) and R. Valent (ESI).
*Engineering System International, France.

References

Berkum, J. V., Figuera, J., Hijne, H. (1989). Computer simulations in an instructional context. Ton de Jon (Ed), The SAFE project, 1-257.

Collins, A., Brown, J. S., and Newman, S. (1989). Cognitive apprenticeship: teaching students the craft of reading, writing, and mathematics. L. B. Resnick (Ed), Knowing, learning, and instruction: Essays in honor of Robert Glazer. Hillsdale, NJ: Erlbaum.

Cooper Eric, B. W., (1991). An architecture for apprenticeship: collaboration with an intelligent tutoring system for qualitative electrical troubleshooting. Lawrence Birnbaum (Ed), Proceedings of the 1991 International Conference on the Learning Sciences, 101-106.

Duffy, T.M., Knuth, R.A. (1990). Hypermedia and instruction: Where is the match? Designing hypermedia for learning. D. Jonassen and H. Mandl (Eds), Heidelberg, Springer-Verlag, 1990, 199-225.

Galdes, D. K., Smith, P. J., (1991). Factors determining when to interrupt and what to say: An empirical study of the case-method tutoring approach. Lawrence Birnbaum (Ed), Proceedings of the 1991 International Conference on the Learning Sciences, 195-202.

Futtersack, M., Labat, J. M., (1993). Dynamic and interactive pedagogical planning, the QUIZ example. AI-ED, Edimbourg, August 24-27.

Futtersack, M., Labat, J.M. (1992). QUIZ: a distributed Intelligent Tutoring System, Proceedings of ICCAL'92, Wolfville, Canada.

Honebein, P. C., Brescia, W. (1992). Applying cognitive apprenticeship to the design of hypermedia environment: The LAB design project. N. Estes, M. Thomas (Eds), The Ninth International Conference on Technology and Education, Paris, France March 16-20, 635-637.

Jong, T., Joolingen, W. V., Hoog, R., Lapied, L., Scott, D., Valent, R. (1993). SMISLE: System for Multimedia Integrated Simulation Learning Environments. DELTA Problem, Barcelonan: April 26-27.

Larsson, J.E., Persson, P. (1988). An intelligent Help System for Idpac, proceedings of ECAI-88, 119-123.

Matthews, M.M., Biswas, G., Neelakandan, H. (1988). USCSH: An active Assistance Interface for Unix, Proceedings of ITS-88, Montréal, 334-341.

Newman, D. (1989). Is a student model necessary? apprenticeship as model for ITS. Artificial intelligence and education, Amsterdam, 177-184.

Sandberg, J., Breuker, J., Winkels, R. (1988). Research on HELP-systems: empirical study and model construction, Proceedings of ECAI, Munich-88, 106-111.

Soloway, E., Guzdial, M., Brade, K., et al. (1992). Technological Support for the Learning and Doing of Design. M. Jones and P.H. Winne (Eds), Adaptative Learning Environments. NATO ASI Series, Vol. F 85, 173-199.

Vivet, M. (1988) Knowledge-based tutors: Towards the design of a shell, International Journal of Educational Research, 12, 8, 839-850.

Vygotsky, L. S. (1986). Thougth and language. A. Kozulin (Ed) Cambridge, MA: MIT Press.

Waern, K. G, (1991). Cognitive aspects of computer aided design. M Helander (ED), Handbook of Human-Computer Interaction, 701-708.

HyperM: A Hypermedia System with Extended Question/Answer Dialogs

G. HOLWEG
Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology, Schieszstattgasse 4a, A-8010 Graz, Austria

P. SAMMER
Institute for HyperMedia Systems (IHM), JOANNEUM RESEARCH, Schieszstattgasse 4a, A-8010 Graz, Austria

Abstract: Most educational software lack either sophisticated CAI-facilities or modern hypermedia functionality. By combining these two aspects we not only increase interactivity between computer and learning persons by means of new functions such as moving dragbuttons across the screen and giving multimedia feedback, but also enhance the educational effect. This paper describes how to embed a system for question/answer dialogs into a hypermedia system and points out the resulting advantages.

During the last decade learning has shifted from rigid system driven courseware to user controlled systems allowing the learning person (in this paper simply called student) to explore the learning material by browsing through and interactively navigating in large databases of information material. Multimedia provides functions to present data in a motivating way stimulating our senses by using different kinds of information material such as films, high resolution digitized photographs and computer graphics, any kind of sound (digitized, MIDI, synthesized speech), or simple text. Hypermedia functions help to organize chunks of information stored in databases by setting links and references. But where is the learning component, someone might ask. With good educational software students should be involved actively, not just simply watching. Learning environments should ask questions and judge the given answers. Many students faithfully believe that they understand what they have just read or heard. But when it comes to applying this knowledge it soon turns out that there is a gap between understanding something and actually applying it. Only when you exercise an operation you can be sure that you have actually understood it. Many popular hypermedia and multimedia systems have been propagated as learning environments, as e.g. HyperCard (Ambron S. & Hooper K., 1990), HyperTies (Shneiderman B. 1989), or Authorware Professional (Authorware 1990) but they cannot often really combine traditional and approved CAI-techniques (CAI stands for Computer Assisted Instruction) with the new features of hypermedia systems such as different kinds of links. The subject of this paper is to illustrate how to build a highly interactive system by merging HyperM - a Windows based hypermedia system - with a powerful system for non intelligent question/answer dialogs (Q/A-dialogs).

HyperM

For several hypermedia projects we needed a powerful PC-based hypermedia system that among other demands should be able to handle huge amounts of data without loss of performance. It should also enable us to add new features.

To this aid, HyperM (Sammer P., 1991) was developed at the IHM. "Images of Austria" is the most famous project running under HyperM. This project was developed in order to present Austria at the EXPO '92 in Seville, Spain, and at the EXPO '93 in Taejon, South Korea in an extended version including Korean text (Maurer H., Sammer P., & Schneider A., 1994). "Images of Austria" consists of some 3000 digitized photographs in high quality which are displayed at remarkably high speed, cartographic material, and digitized film-clips which can be played without additional hardware. Some 18000 textual supplements are available. The user can select one of the languages English, French, German, Italian, Korean, and Spanish in order to get all the language-sensitive information including the user interface in the desired language.

The user interface can easily be redesigned for specific application to fit needs such as self explanatory and intuitive navigation support (e.g. for going back to the previous scene, scrolling large images, or branching to different parts of the presentation).We did not use all of the numerous features available in HyperM for the EXPO projects in order to obtain the necessary "lack of complication". The database management system (DBMS) for example allows complex database manipulations using SQL (structured query language). As a result of the flexibility of our hypermedia system we could not only incorporate peripheral devices (e.g. starting the presentation triggered by the beam of a photoelectric barrier) but also complex modules such as sophisticated answer judging.

The question/answer system

The Q/A-system has been designed as an independent module which allows the creation and execution of topic independent, highly interactive Q/A-dialogs. This section gives a general view of the Q/A-system and explains how it has been embedded into HyperM. A detailed description of all features can be found in Holweg (1994), implementational details can be looked up in Spinka (1993) and Dick (1994).

A Q/A-dialog is an interactive process where the system presents problems by asking questions and the student solves the problems by answering the questions. Problem solving should be done as intuitively as possible by clicking on buttons, moving graphical objects, entering text, or any other kind of user input you can think of (as for example input from a MIDI-keyboard, data gloves or touch screens). Our system can currently handle input from keyboards and from pointing devices. Each type of user input has its typical characteristics which can be checked (e.g. text input can be compared with a predefined model answer, the movement of graphical objects can be compared with predefined moving paths). Based on ideas of object-oriented structures our Q/A-system consists of a number of objects, each representing a type of input as well as rules how to judge the student's input. These objects are called **interactors** (IAs for short) because they have the ability to receive user input, judge it, and react according to the results of judging. In this way interactivity between system and student is guaranteed. The Q/A-system can be easily extended by adding new interactor types, as for instance an interactor that handles the input of a MIDI-keyboard. However, a single interactor is too restricted in its features to construct complex Q/A-dialogs. Therefore a Q/A-dialog consists of several interactors which are supervised by the **control unit**. The control unit contains a predefined flow structure of a Q/A-dialog coordinating when an interactor can be used by the student and when it is locked (Figure 1 schematically shows how interactors, judging algorithms, reaction parts and control units work together. Figure 3 gives an example of a Q/A-dialog).

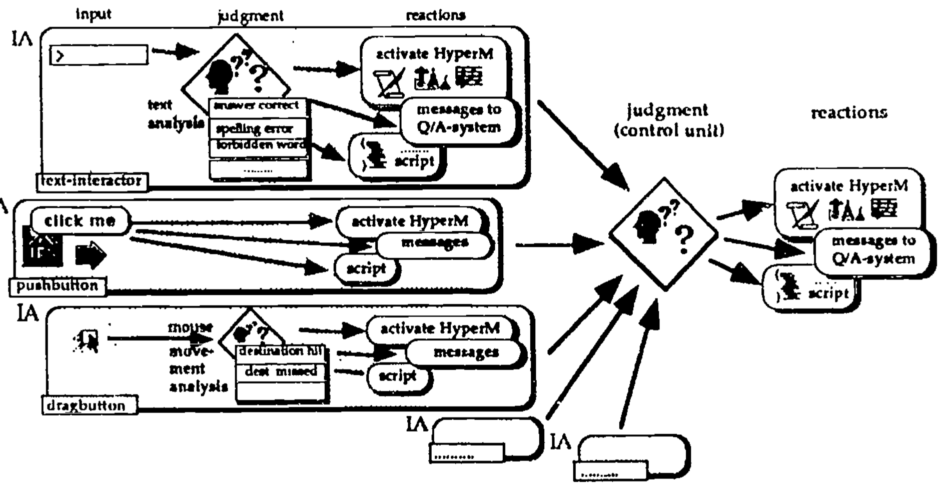


Figure 1: Q/A-dialogs - a cooperation of interactors, judging algorithms, reaction parts and control units.

As already mentioned the activity of an interactor deals with three essential points: user input -> judgment -> reaction. The Q/A-system is designed as a platform independent module that can be easily embedded into different hypermedia systems. In contrast to the judgment section which is implemented as a totally independent module there are parts of the user input and reaction section that are tightly connected to the superior hypermedia system and its user interface. Hence processes such as entering a text-answer or clicking on buttons, and reactions as e.g. displaying a picture on the screen, activating audio signals, or giving a textual response have been removed to the superior system. Figure 2 schematically shows how the Q/A-module is embedded into HyperM. Although as a consequence, this method has some disadvantages (HyperM has to know about at least the input features of the Q/A-module) the advantages are prevailing. The Q/A-system can react in any way by just requesting HyperM to act. The command "*activate subscene name_of_subscene*" directs HyperM to execute a subscene with the specified name. A subscene is similar to a subprogram in programming languages which has access to all features of the language. Thus, the Q/A-system has access to all features of HyperM including all "Hyper-functions". In the following we have listed some of the points which result from a combination of the two systems:

- The judging of a student answer causes HyperM to branch to specified locations and can thus be used to navigate in the learning material.
- Different languages are supported. Text messages and responses can be edited in several languages. HyperM automatically displays the text in the selected language.
- Hypertext features can be used. Words in text messages and responses can be clicked on and links to dictionaries, encyclopedias or other parts of HyperM can be executed. However, authors of Q/A-dialogs have to be careful when formulating questions while this feature is allowed. A question that asks for the birthday of Napoleon might for example encourage the students to look up the answer in an encyclopedia just by clicking on the word Napoleon instead of answering the question themselves.
- Keywords attached to Q/A-dialogs make those dialogs accessible through queries. Thus libraries of Q/A-dialogs of different topics can be collected.

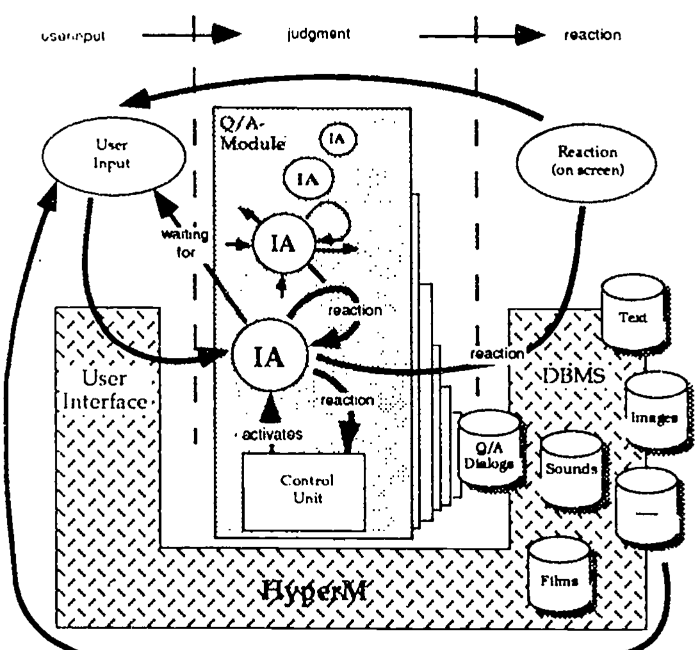


Figure 2:
The Q/A-module is responsible for controlling a Q/A-dialog and judging the user input. User input itself and reactions such as displaying a picture, playing sound or giving text responses are done by the superior hyper-media system.

The Q/A-system currently supports the following interactors: *TextInput*-fields, *PushButtons*, *RadioButtons*, *TouchButtons*, *DragButtons*, *CheckBoxes* and others which do not get their input from the student (*Timers*, *Counters*, *Conditionals*). These "other" interactors are important for controlling the internal flow of Q/A-dialogs by performing reactions under special conditions (such as time-outs or counting the number of wrong answers). Interactors can be arranged in any desired order. Figure 3 shows a typical example that extensively makes use of *DragButtons* which can be clicked on and moved across the screen. The left part of this figure shows a sketch of a bicycle brake. After the brake has been displayed to the student, it is erased from the screen and the student has to reassemble the brake by dragging each single part with the mouse to its corresponding position (right part of the figure). All parts are defined as interactors of type *DragButton* and have one or more destinations where they can be moved to (a bolt can for example be used either on the left or the right side of the brake, therefore the bolt has defined more than one destination). The order of assembling is not arbitrary, a nut for example cannot be set before the bolt has been fixed. Thus, the student has to take care of special rules (as they occur in the daily work of a mechanic). These rules have to be specified in the judging section of the *DragButtons* and in the control unit performing the task of supervising the learning person. The following list shows some examples of such rules:

- DRAG aluminum part 1 TO position aluminum 1
- DRAG bolt1 TO (position bolt left OR position bolt right)
- start with aluminum part 1 or aluminum part 2 in any order
- bolt1 must not be moved before aluminum part 1 or aluminum part 2 has been set

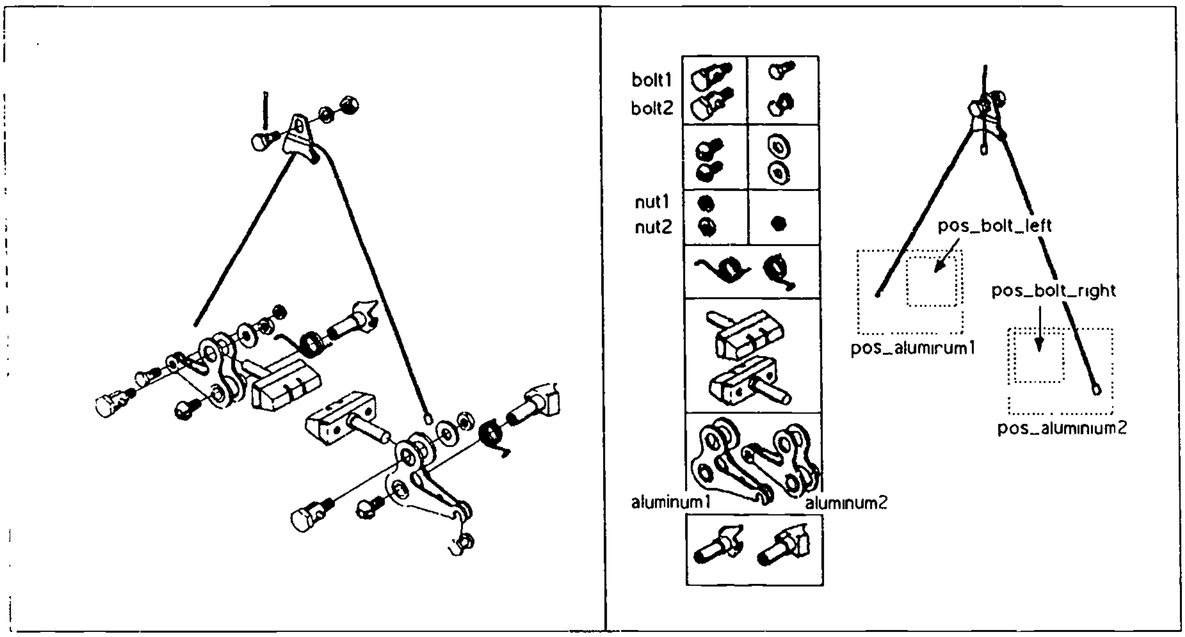


Figure 3: This is an example of a bicycle brake that has to be assembled by snapping each part with the mouse and dragging it to its corresponding position on the right side of this figure.

Components of the Q/A-system

From the technical point of view Q/A-dialogs consist of interactors and control units. But there are other properties that are responsible for a flexible and powerful use of Q/A-dialogs. This section presents some points that were considered to be important for the development of our Q/A-system.

- The power of every Q/A-system stands and falls with the availability of flexible *Judging Algorithms*. The user input must be examined for correctness by comparing the given answer with a predefined model answer. Most interactors typically offer judging algorithms as part of their features. A *TextInput*-interactor for example contains a judging algorithm that checks student answers for spelling errors, synonyms, forbidden words, repeated words, word groups and intervals of numbers, date or time. To find spelling errors we have used the popular algorithm for the *Weighted Levenshtein Distance* (a short description can be found in Nesbit (1986)) because this algorithm is easy to use and returns a measure for the dissimilarity of the compared words. *DragButtons* use a judging algorithm that watches over the student's movements with the mouse. The learning person has to move the *DragButton* to predefined destinations, touch areas on the screen, follow given paths or stay within a given area. The control unit itself is nothing else than a judging algorithm that supervises the flow of a Q/A-dialog. Because of its modularity the Q/A-module can be easily enhanced by new judging algorithms and existing algorithms can be exchanged by more powerful ones. In this way judging text answers could be managed by language understanding systems based on semantic analysis (which in contrast to the existing method requires language dependent knowledge based databases) (see e.g. Crocker M.W. & Lewin I., 1992). Methods of artificial intelligence as those used in game playing theory (Winston P.H., 1984) could be incorporated into the control unit and thus help to decide if students' actions have to be judged right or wrong.

- When we consider Q/A-dialogs from the pedagogical point of view, we see a need for several *Learning Strategies*. There exists a number of standard types of learning strategies such as multiple choice tests, vocabulary tests, drill&practice and form filling which are well known features in computer assisted learning (see e.g. Merill P.F. & Salisbury D., 1984; Huber F., Maurer H., & Makedon F., 1988; and Nesbit J.C. & Yamamato N., 1991). They are still interesting nowadays especially when they are mixed with graphical features. A multiple choice question for example could be answered by simply clicking on the correct answers. However, they are often regarded as old-fashioned because of their rigid behaviour. In addition to these well known learning strategies it is necessary to offer strategies that have shifted the control from the system to the learning person. The student can experiment with a problem by trying different solutions thus figuring out which one is correct. Most times there is not only one correct solution to a problem, there can be many ways to finish a Q/A-dialog in the right way. The example with the bicycle brakes is typical for that kind of strategy which can be placed somewhere between *Learning by Doing* and *Learning by Trial and Error* (see e.g. the articles of Schank (1993) who presents an interactive multimedia system for learning and Vivet (1991) who deals with those strategies from a pedagogical point of view). New strategies can easily be created by use of the control unit, approved stategies can be stored in databases and accessed whenever needed (see also the following paragraph about databases).

- Sometimes it is useful to have algorithms that repeatedly ask the same type of question but use different parameters and values. We call them *Question Asking Agorithms*. Questions and answers are not exactly defined by the author, but they are established at execution time. This type of Q/A-dialog is known as *adaptive CAI* in literature (see Uhr L., 1969; or Park S. & Tennyson R., 1983 for a survey of early systems). A vocabulary test for example gets words and translations out of a database. The selection of vocabularies depends on special learning strategies such as the *n-piles method* (Stubenrauch R., 1989) which repeats incorrectly answered questions. Linear equations and definite solutions can be calculated at runtime which guarantees alternation since the same question can contain different variables each time it is presented (see Maurer H., Stone M.G., Stubenrauch R., & Gillard P., 1991 for a discussion on the pros ans cons of such algorithms).

- HyperM extensively uses *Databases* for storing all kinds of information such as images, sounds, films and last but not least Q/A-dialogs. This guarantees easy access to the data and makes enlargements simple. Questions can be collected in databases and can then be accessed by HyperM through keywords and queries. Thus it becomes possible to ask HypeM to post all questions about a specific topic, as for example sorting algorithms. The Q/A-system itself uses databases. Vocabulary tests for instance get words and translations out of a database. The order of questions depends on question asking algorithms as we mentioned above. Furthermore, databases can be useful for editing Q/A-dialogs. Standard types of learning strategies, control units, predefined interactors and model answers are best qualified for being stored in databases and being accessed when needed.

Conclusion

Most of the components listed above have already been implemented and we are working on the completion. We have tried to combine the advantages of a hypermedia system with the features of a Q/A-system. Information and learning material can be presented in a modern way by displaying all kinds of multimedia data. The user can browse through the presented material by following links and exploring the contents of the presentation. The Q/A-system offers both traditional learning strategies such as drill&practice and form filling, and Q/A-dialogs where the control is partly given to the learning person and the system's job is to watch over the student's activities. The learner should be able to focus on the contents and not the operation of the program, problems should be solvable intuitively, according to the motto: "you don't explain the solution to a problem, you just show it".

References

Ambron S. & Hooper K., (1990). *Learning with Interactive Multimedia*, Microsoft Press.

Authorware (1990). *Authorware Professional Reference Manual*, Authorware Inc., Minneapolis MN.

Crocker M.W. & Lewin I., (1992). *Parsing as Deduction: Rules versus Principles*, Proc. of the ECAI'92, Vienna, pp. 508-512.

Dick H. (1994). *Integration eines Frage/Antwort-Systems in HyperM*, Diplomarbeit, Graz Univ. of Technology, is to appear.

Holweg G. (1994). *Aspekte von Frage/Antwort-Dialogen in Multimedia- und Hypermedia-Systemen*, Dissertation, Graz Univ. of Technology.

Huber F., Maurer H., & Makedon F. (1988). *HyperCOSTOC: A Comprehensive Computer-based Teaching Support System*, IIG Report 258, Graz Univ. of Technology.

Maurer H., Sammer P., & Schneider A. (1994). *Multimedia Systems for the General Public: Experiences at World Expositions and Lessons Learned*, Proc. of IIMS'94, Perth, Australia, pp.333-341.

Maurer H., Stone M.G., Stubenrauch R., & Gillard P.(1991). *Question/Answer Specification in CAL Tutorials (Automatic Problem Generation Does Not Work)*, Didaktik der Menschheit 21, Hölder-Pichler-Tempsky, Wien, B.G.Teubner Stuttgart, pp.191-197.

Merill P.F. & Salisbury D. (1984). *Research on Drill and Practice Strategies*, Journal of Computer-Based Instructions, 11(1), pp. 19-21.

Nesbit J.C. & Yamamato N. (1991). *Sequencing Confusable Items in Paired-Associate Drill*, Journal of Computer-Based Instruction, 18(1), pp. 7-13.

Nesbit J.C. (1986). *The Accuracy of Approximate String Matching Algorithms*, Journal of Computer-Based Instruction, 13(3), pp. 80-83.

Park S.I. & Tennyson R.D. (1983). *Computer Based Instructional Systems for Adaptive Education: a Review*, Contemporary Education Review, 2(2); pp. 77-98.

Sammer P. (1991). *Hypermedia Basis System*, Proc. of 6th Austrian Hungarian Conference on Intelligent Systems CIS'91, pp.281-289.

Schank R.C. (1993). *Learning via Multimedia Computers*, Communications of the ACM, 36(5), pp. 54-56.

Shneiderman B. (1989). *Reflections on Authoring, Editing, and Managing Hypertext*, The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, Cambridge MA: The MIT Press; pp.115-131.

Spinka O. (1993). *Antwortanalyse in HyperMedia-Systemen*, Diplomarbeit, IICM, Graz Univ. of Technology.

Stubenrauch R. (1989). *Advanced "Non-Intelligent" Question-Answer Dialogs*, IIG Report 268, Graz Univ. of Technology.

Uhr L. (1969). *Teaching Machine Programs that Generate Problems as a Function of Interaction with Students*, Proc. of the 24th National Conference; pp.125-134.

Vivet M. (1991). *Learning Science & Engineering with Open Knowledge Systems*, proceedings CALISCE91, Lausanne Switzerland, pp. 53-62.

Winston P.H., (1984). *Artificial Intelligence*, Addison-Wesley.

TurtleGraph: A Computer Supported Cooperative Learning Environment*

JIHN-CHANG J. JEHNG
Institute of Educational Media & Library Sciences
Tamkang University, Taipei, Taiwan, R.O.C.
E-Mail: tkut067@twnmoe10.bitnet

YU-FEN SHIH
Department of Information Management
Ming-Chuang College, Taipei, Taiwan, R.O.C.

STEVEN LIANG & TAK-WAI CHAN
Institute of Computer Science and Electronic Engineering
National Central University, Chung-Li, Taiwan, R.O.C.

Abstract: The paper discusses a computerized learning environment called TurtleGraph that is designed and developed to support collaborative problem solving. Within the learning environment, learners are requested to write computer programs to generate geometric figures. The instructional focus of the system is to enhance the learner's LISP recursive programming skill by making strategic thinking more explicit, inducing reflection through reciprocal evaluation and criticism, and fostering an active role of learning through collaboration. A conversational model is formulated to address the role of knowledge in the collaborative problem solving process, and the design of the system is mainly guided by the theoretical model. In addition to the model, several instructional design principles are also incorporated to make the learning environment be more educationally effective and efficient.

Theoretical Framework

Influence of peer collaboration on individual cognitive development has been investigated by a number of educational and psychological researchers, and findings of previous research have provided us rich information in this regard. First, collaborative problem solving could be an effective catalyst for the enhancement of individual cognitive skills. Piaget (1965) indicates that peer learning could reduce egocentrism and provide scaffolding for cognitive development. Alternative to this view, Vygotsky (1978) proposes the concept of the zone of proximal development that social interaction plays an important role in shaping individual cognitive structures. The development of individual cognitive skills could be determined through problem solving in collaboration with more capable peers. Second, collaborative problem solving may appeal more intensity of learner's involvement and attention to the learning activity. Collaborative problem solving activities usually are intrinsically motivating to the extent that they engage individuals in a process of seeking to solve problems and accomplish goals that require the exercise of valued personal skills, such as the ability of self-determination and interpersonal skills (Lepper, 1985). Third, collaborative problem solving is important for inducing individual metacognitive skills. Participants in a group could learn to control and monitor their behaviors, and to develop more effective problem solving strategies through reciprocal evaluation and criticism (Palinesar & Brown, 1984).

Over the last several years, researchers have shifted the focus and become intrigued by the reciprocal relations between individual cognition and social interaction (Resnick, Levine, & Teasley, 1993). Rather

* The research project is under the group project entitled LISA, which is currently funded by the National Science Council, Taiwan. LISA (Learning IS Active) is a group project with social learning as the central focus. The goal of the project is to establish a multi-channel learning envrionment, where a student engages in social learning and interacts with various agents who may be human beings or computer simulated agents. A dozen of researchers and graduate students from different universities in Taiwan are involved in the LISA project.

than seeking to understand cognitive and social processes in isolation or treating one process as context for the other, a growing number of studies seek to provide conceptual schemes that allow the investigation of thinking as socio-cognitive activity. Particularly the development of the situated and social learning theory has allowed us to rethink the nature of human learning and problem solving. Learning is considered as the development of socially shared cognition (Brown, Collins, & Duguid, 1989). Knowledge is no longer merely regarded as stored representation of acquired experience. Knowledge is always a novel construction that continues to develop in every action in relation to the practice of a community (Clancey & Roschelle, 1991; Clancey, 1992).

The dynamic view of learning as continuous alteration of an individual's current knowledge state has certain instructional implication. In a typical teacher-student instructional situation, the goal of a teacher's work can be regarded as providing support to alter the student's evolving knowledge state so that it may converge to his own. An important function that the teacher's exercise is to monitor the learning activity and assist the student so that the convergence goes effectively (Chan, 1991). An analogous process can also be used to depict the collaborative problem solving. Participants in a group learn to construct knowledge by mutually examining, monitoring, assessing and justifying each other's knowledge. The more capable learners play a role of cognitive supporter to the less capable ones. A successful collaboration occurs when different versions of knowledge merge into one that is conducive to the solution of the problem.

In our preliminary study, subjects were arranged in pair and were requested to write LISP codes to solve some geometric figure problems. In order to explore the structure of a successful collaborative problem solving, the subjects' conversational dialogues were recorded and analyzed. By examining the dialogues, we found that the role of knowledge is critical to a successful collaboration. A conversational model was developed to capture its importance. In this model, we roughly partition a collaborative problem solving process into three knowledge construction stages, and they are knowledge communication, knowledge negotiation, and knowledge consolidation stage. It is argued that in the knowledge communication stage, participants in a group try to acquire as much problem information as possible. They merely exchange and transmit information to make the problem as clear as possible. Communication of knowledge requires least coordination of social interaction and involves little maintenance of shared conception. As participants acquire more problem information and understand each other better, they will be more involved and start to make contribution to the collaboration. At the stage of knowledge negotiation, an individual begins to formulate problem solutions and attempts to share his ideas with his partner as partial solutions are developed. He may make efforts to explain his ideas or seek to draw evidence to convince his partners while disagreement occurs. Proposals to the problem solution continue to be refined and be more consonant as more dissonance is resolved. Knowledge negotiation requires certain degree of coordination of participants and involves more maintenance of shared understanding. In the knowledge consolidation stage, participants of a group continue to collaborate to construct and maintain a shared conception of a problem. They work together to examine and validate their solutions. Knowledge consolidation is highly situated and requires the highest degree of coordination among the partners. The design of our collaborative learning environment is on the basis of this theoretical model. Figure 1 shows the three knowledge construction stages.

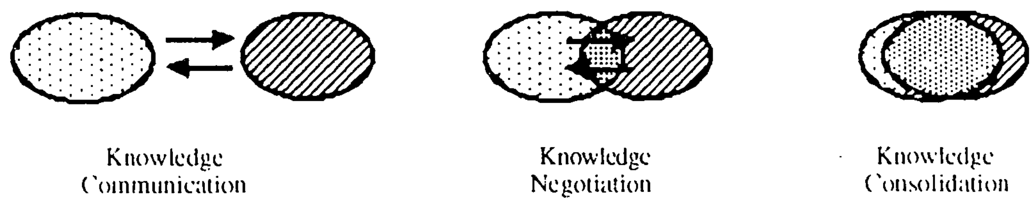


Figure 1. The conversational model

Since knowledge is critical to a successful collaboration, it is believed that cooperative learning environment must be designed in a manner that enables group members to communicate and share their ideas effectively. Baker (1991, 1992) indicates that an intelligent teaching system could provide some guidance in conflicting situations to facilitate student's explanations within a collaborative context. In order to provide guidance to facilitate knowledge construction, the development of our learning system uses a deliberately-designed restricted communication interface to induce individual meta-communication skills Within the learning environment, the student is encouraged to express their thoughts explicitly by

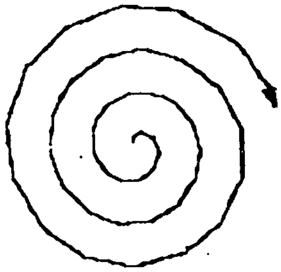
* identifying the goal of the problem,
* clarifying conflicting arguments,
* elaborating the supporting reason of explanatory propositions, and
* elaborating the opposing reason of argumentative propositions.

It is argued that a learning system that promotes meta-communication skills could help students maintain shared cognition, carefully form their argumentation, reduce ambiguous propositions, and construct thoughtful explanation. A successful collaborative problem solving activity can then be easily accomplished.

Principles Of Instructional Design

The LISP recursive function, that is taught at the introductory level of computer science courses, is the subject selected to explore the design and development of the cooperative learning environment. For students with computer science major, understanding the LISP recursive function is fundamental to the advancement of higher level of computer science courses. However, the student always have a hard time to learn to write the LISP codes, since little knowledge can be transferred from their previous experience about the procedure-based language, such as PASCAL and C. The goal of our system is to train students who have learned the syntax of LISP language with little LISP programming experience.

In the TurtleGraph, the student will learn the specific nature of the recursive function by programming LISP codes in combination with several simple LOGO commands. LOGO is a language developed by Papert and his colleagues in MIT, and it is used to teach students about the notion of computer programming and some deepest ideas of science and mathematics (Papert, 1980). In the TurtleGraph, the task is designed in a manner that the student is requested write codes that move the turtle at the pre-designated direction and angle to draw a geometric figure. While drawing the geometric figure and tracing the movement of the turtle on the screen, the student can readily capture the profound nature of the recursive function. By situating the learning activity in the problem solving context, the system can help the student understand the specific feature and function of acquired knowledge. Figure 2 shows a spiral form of geometric figure generated by the TurtleGraph codes.



```
(defun spiral ( counter length angle )
  ( cond
    ( ( /= counter 0 )
      ( fd length )
      ( rt angle )
      ( spiral ( - counter 1 ) ( + length 1) angle )
    )
  )
)
```

Figure 2. A spiral form of geometric figure with TurtleGraph codes.

Several instructional principles are also incorporated in the system to make each learning activity be more educationally sound.

A Controlled Conversational Environment. The design of the TurtleGraph applies the "button control" strategy (Schank & Jona, 1991) to help the student regulate the conversational protocol so that it can promote more effective and efficient collaborative problem solving. Each time when the student presses a conversation button to make communication with his partner, the button prompts him to elaborate his conversational statements depending on the type of conversation button selected. It may request the student to elaborate what he is trying to propose, or the reason why he agrees or disagrees with his partner's ideas. The button is designed to assist problem solvers to carefully develop their ideas and regulate their conversational behaviors during the collaboration. All the conversational dialogues are recorded and displayed in sequence on the computer screen. The student can utilize the dialogue protocol to (a) understand the interaction, (b) reflect his own problem solving behaviors, and (b) organize the conversational action and sequence. With the assistance of the conversation button, students could hold one another accountable for meaningful participation in a collaboratively sustained problem solving situation.

A Conceptually Visualized Learning Environment. In the TurtleGraph, geometric figure problems are used as means of teaching the concept of LISP recursive function. It is a display-based problem solving environment that emphasizes representing problems in an external display (Larkin, 1988). Within the TurtleGraph, the student can immediately see and reflect on the process of how geometric figure is generated by the turtle while manipulating the LISP codes. The external display can help the student explicitly envision his current knowledge state and help him closely examine, analyze, and modify his knowledge. External display can also reduce the student's cognitive load by making the knowledge acquisition and interpretation processes perceptual, rather than logical. Processing information perceptually is easier for the novice learner than processing information logically. Moreover, arguments among participants in a group can easily be resolved through perceptual comparisons and judgments than through logical inferences from purely nonvisual facts.

A Structured Problem Solving Environment. In the TurtleGraph, various geometric figure tasks were designed in accordance with three different types of recursive function. The three types are head, middle, and tail recursion. The classification is based on the location of recursion in the program. A good number of representative problems were sampled with different complexity and difficulty. Each problem begins with a story and the student is assumed to play the role of computer graphic designer inside the story. Story-based problems help the student to envision how the knowledge could be used in realistic situations (Cognition and Technology Group at Vanderbilt, 1990, 1992). Just as an effective teacher pays close attention to a student's current level of ability and gives his tasks that build on his prior learning, the TurtleGraph is a structured practice environment that sequences learning experience based on each individual student's problem solving performance. Each time a student accomplishes a task, the TurtleGraph will assess the student performance and make a decision to give an easier or a more difficult problem to the student for next learning. The student must master a certain level of problem solving skill before advancing to attack another level of task.

The Architecture Of The System

The system is coded by the SuperCard application and the Lisp interpreter, and is run in the Macintosh work station. At the first section of the system, a tutorial is provided to teach the student several simple LOGO commands that are used to draw figures by moving the turtle on the screen. As soon as the student finish the tutorial, he will read a story containing the problem. By tracing and analyzing the student's prior performance history, the system will select a problem that corresponds to the student's current ability level. If the student chooses to attack the problem, the system will bring him to the TurtleGraph Cooperative Problem Solving Environment, where he can choose to solve the problem alone or seek a partner to collaborate with. The environment contains five instructional areas, including the Control Panel, the Turtle Window, the Dialogue Recorder, the Program Editor, and the Listener. Figure 3 shows the TurtleGraph Cooperative Problem Solving Environment.

The Control Panel contains two groups of control button: system and communication button. All learning activities are operated by the system button. The Help button contains general help information that could guide the student to use the system. The student can press the help button at any time to request assistance from the system. It is designed in a manner that is sensitive to the student's need. Each time the student presses the Description button, the system will provide detailed problem information. The Example button contains example programs for student's reference. It is designed to help the student organize his programming knowledge and develop problem solving strategies. The student can run the example programs and also can copy and adapt them for his own use. Within the TurtleGraph, the student can choose to solve a problem independently or seek to collaborate with other students. Once the student presses the Commu button, all the communication buttons become activated, and the student can start to work with his partner. By pressing the Check button, the system will judge the student's solution. The Check button also provides the Answer option if the student wants to find out the correct solution. The student can press the Exit button to quit the system and leave it at any point, and the system will record his performance status and update his performance history in the database. The TurtleGraph will provide instruction and suggestion at the end of each learning session.

All communication buttons are designed to support the collaborative problem solving. The student can press any communication button to discuss with his partner during the problem solving. Once the student presses the button, the system shows a dialogue box prompting him to either make a clear statement about what he is trying to do or elaborate the reason why he agrees or disagrees with his partner. The prompt is designed to help the student monitor his problem solving behavior, regulate the conversational protocol and make an effective collaboration with his partner

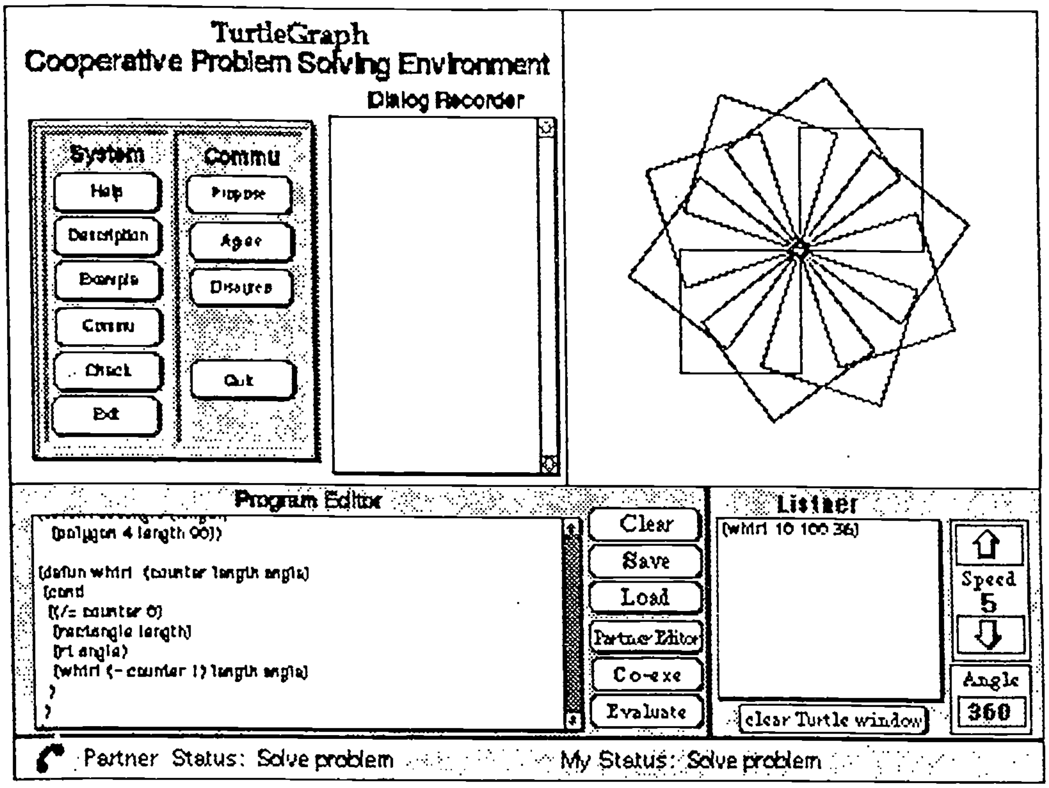


Figure 3. The TurtleGraph Cooperative Problem Solving Environment.

The Turtle window contains a turtle for drawing. Inside the window the screen will show the figure with the movement of the turtle immediately after a program is executed. The Dialogue Recorder keeps track of all the dialogue statements in sequence made by the problem solvers during the communication, and all statements can be saved for future reference. Each dialogue statement in the Dialogue Recorder is marked with the communication button that the student just pressed to highlight the content of the statement. The student can read and reflect on the dialogue statements during the collaboration.

The lower part of the screen is the Program Editor and the Listener. The student generates the LISP codes inside the editor window. The editing function like copying, cutting, and pasting can be accomplished by pressing the pre-defined shortcut keys. The purpose of the save and load button is for saving and retrieving programs. The student can take a look at his partner's program by pressing the Partner Editor button. Another special design in the Program Editor is the Co-exe button. The student can press Co-exe button to execute his program and the figure that the program generates will simultaneously appears in his and the partner's Turtle Window. The design of the Partner Editor and Co-exe button aims to facilitate the process of collaboration.

The student can evaluate his TurtleGraph codes at any time inside the Listener box. He can also press the Speed button to control the speed of Turtle movement. The Angle Box shows the degree of the angle when the turtle make a turn. At the bottom of the screen the system shows the message of the collaborative status. All facilities provided by the TurtleGraph Cooperative Problem Solving Environment fully support the collaborative problem solving activity.

Discussion

This paper presents an ongoing research project that has designed and developed the TurtleGraph collaborative learning environment. With the restricted communication interface, the system can prompt students to make more effective collaboration by reducing the ambiguity of the explanation, elaborating the augmentative reasons, organizing the dialogue sequence during the conversation. In addition we plan to make greater efforts to include AI techniques to expand the capability of the communication facility and the answer diagnosis. The technical infrastructure of the TurtleGraph cooperative learning environment will continue to develop with more thorough evaluation.

Acknowledgments

The authors gratefully acknowledge the valuable suggestion made by Dr. Michael Jacobson in the final draft of this paper.

References

Baker, M. J. (1991). The influence of dialogue processes on students' collaborative generation of explanation for simple physical phenomena. In *Proceedings of the International Conference on the Learning Sciences* (pp.9-19).

Baker, M. J. (1992). Modeling negotiation in intelligent teaching dialogues. In M. Elsom-Cook, R. Moyse (Eds.), *Knowledge negotiation* (pp.191-240). London, UK: Paul Chapman Publishing.

Brown, J. S., Collins, A., Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher*, 18(1), 32-42.

Chan, T. W. (1992). Integration-Kid: A learning companion system. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence* (pp. 1094-1099). Sydney, Australia: Morgan Kaufman Publisher, Inc.

Clancey, W. J. (1992). Guidon-Manage revisited: A socio-technical systems approach. In *Proceedings of Intelligent Tutoring Systems '92*. Berlin, Germany: Springer-Verlag.

Cognition and Technology Group at Vanderbilt. (1990). Anchored instruction and its relationship to situated cognition. *Educational Researcher*, 19(6), 2-10.

Cognition and Technology Group at Vanderbilt. (1992). The Jasper experiment: An exploration of issues in learning and instructional design. *Educational Technology: Research and Development*, 40(1), 65-80.

Larkin, J. H. (1988). Display-based problem solving. In D. Klahr & Kotovsky (Eds.), *Complex information processing: The impact of Hebert A. Simon* (pp. 1-39). Hillsdale, NJ: Lawrence Erlbaum.

Lepper, M. (1985). Microcomputer in education: Motivation and social issues. *American Psychologist*, 40(1), 1-18.

Palincsar, A. S., & Brown, A. L. (1978). Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities. *Cognition & Instruction*, 1(2), 117-175.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books, Inc.

Piaget, J. (1965). The moral judgment of the child. New York: The Free Press.

Resnick, L. B., Levine, J. M., & Teasley, S. D. (1991). *Perspectives on socially shared cognition*. *Washington*, DC: American Psychological Association.

Roschelle, J., & Clancey, W. J. (1991). *Learning as social and neural*. Paper presented at the 1991 Annual Meeting of the American Educational Research Association at Chicago.

Schank, P., Linn, M. C., & Clancy, M. (1990). *How does an on-line template library help students learn PASCAL?* Paper presented at the 1990 Annual Meeting of the American Educational Research Association at Boston.

Schank, R., & Jona, M. Y. (1991). Empowering the student: New perspectives on the design of teaching systems. *The Journal of Learning Science*, 1(1), 7-35.

Vygotsky, L. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press. (Original work published in 1935.)

A Re-usable Algorithm for Teaching Procedural Skills

MARK K. JONES, ANDREW S. GIBBONS
Department of Instructional Technology
Utah State University, Logan Utah 84322-2830

DENISE C. VARNER
Southwest Research Institute
6220 Culebra Road, San Antonio, TX 78228-0510

Abstract: The design of a re-usable instructional algorithm for computer-based instruction (CBI) is described. The algorithm is designed to reduce development and life cycle costs for CBI by providing an authoring environment suited for subject matter experts who do not have instructional skills, and by supporting rapid prototyping. The specific algorithm described implements a simulation-based reactive environment for learning and practicing device operation skills. The approach to designing and implementing the algorithm is general and should apply to other instructional outcomes.

Problem Statement

The opportunity for repeated practice is critical to the development of procedural skills. The effectiveness of practice may be increased by providing guidance and feedback to the learner and by structuring the practice environment to transition in small steps from simple to complex practice situations.

Utilizing a computer to provide practice environments has several advantages. In comparison with the real environment, a computer may be more convenient, safer, and less expensive. In comparison with human-based instructional methods, a computer is more patient, more consistent, and available at a time and location convenient for the individual learner. In comparison to other media-based instructional methods, a computer may provide a reactive practice environment and the potential to customize the instruction to the individual learner.

A reactive practice environment is key to learning procedural skills. The learner performs and the environment reacts. The reaction approximates the effect the learner's action would cause in the real environment. This provides context for the learner and prepares for the transition from the practice to the real environment. The realistic reaction may be augmented by explicit instructional guidance and feedback that would not occur outside the instructional setting. Both learner and the instructional control program may vary the conditions of the environment and the requirements for performance based upon the learner's progress.

Unfortunately, developing computer-based instruction which incorporates a reactive environment, explicit guidance and feedback, and variable practice conditions and performance requirements, is very expensive. Highly specialized personnel are required. The high cost and long timelines of development and the scarcity of skilled personnel limit the application of this type of instruction. It is simply not cost-effective if the potential audience is small, the instruction will have a short useful life, or the development is being undertaken by a small company or intended for public education.

The goal of our research is to dramatically lower the cost of development and the required skill level of development personnel. The approach is to provide re-usable instructional programs in which the strategy and tactics are pre-defined for a particular instructional objective. The instructional developer need only describe the desired performance and the environment of the performance.

Approach

Existing computer-based instruction (CBI) authoring tools successfully address a specific problem. It is assumed that the author, someone with teaching or instructional design experience, knows what they want to teach and how they want to teach it -- in fact they may already be teaching the material using non-computer means. The author is not however an experienced computer programmer, and does not know how to program the computer to deliver the desired instruction. The authoring language or system is designed to help the author make the translation from what to teach and how to teach, which is known, to how to make the computer teach, which is unknown, and to accomplish this without requiring the author to become an expert programmer.

As computers have become more widely available, more organizations have desired to develop computer-based instruction than there are available teachers and instructional designers. It is now commonly the case that the person responsible for developing CBI is not only not a programmer, but is not an experienced teacher or instructional designer either. The skill this person is most likely to have is subject matter expertise. The assumptions made by existing authoring systems -- that the author knows what to teach and how to teach it -- are no longer valid. In fact, lack of knowledge in these two areas may be the most important barrier to the development of effective CBI.

The successful solution to the problem of teachers who did not know how to program computers was not to teach them to be programmers, but rather to provide tools that allowed them to create CBI without needing to learn much about programming. Similarly, the solution to the problem of subject matter experts who do not know much about instruction will not be to teach them to be teachers, but to provide tools which allow them to create CBI without needing to learn either programming or instructional design.

This approach does not focus on making tools that are just "simple to use". Teaching and learning are complex human endeavors with many variables and incomplete knowledge. This complexity cannot be entirely avoided, however the nature of the complexities presented to authors can be managed. The most important step that can be taken is to assure that the authoring decisions, whether simple or complex, require only knowledge that it is reasonable for the author to have. In other words, the context of the authoring should be familiar to the author. In the case of a subject matter expert, it is feasible to require authors to describe the desired performance and the objects upon which that performance will occur. While some authoring tasks may require authors to think more deeply about these and to describe them more explicitly than they have before, the domain is at least familiar and accessible. On the other hand, even fairly simple authoring tasks that require instructional or programming expertise are inaccessible to the average subject matter expert.

Instruction

We have developed a prototype re-usable instructional algorithm. The prototype is implemented on IBM PC compatibles running the Windows™ graphical environment, using the prototyping tool ToolBook™.

The algorithm is designed to instruct algorithmic procedures in the area of device operation. Examples would include the operation of many electronic or mechanical devices. The specific test case for the prototype was the performance of the Central Air Data Computer Self-Test procedure for F-16 aircraft.

The algorithm begins with a media presentation. This, and all similar media presentations, is completely under the control of the author and may range from a piece of text or a graphic to a complete multi-media presentation including digital audio and video.

The lesson then continues with a short tutorial. This is designed both to introduce the procedure and to acquaint the student with some of the learner control capabilities. The learner is shown general information about the procedure and the sequence of steps that make up the procedure. The results dialog box, which will display the learner's progress, is introduced. The learner is also shown a list of all personnel involved in the procedure and (under author control) may select which role to practice. Each set of information is presented in a dialog box that is keyed to one of the icons in the toolbar. After this introductory tutorial, only the results dialog will be automatically displayed again, however all dialogs are available at the learner's discretion throughout the instruction.

After the tutorial, the heart of the instruction commences. This is organized around a set of performances. The author may add, subtract, and customize the performances, but by default there are five: a demonstration, three levels of practice, and a self-check.

Each performance instructs all steps of the procedure. The algorithm is designed to instruct the performance of each step, the sequence of steps, who performs each step, and any tools that are required for the performance of a step In addition, the algorithm connects the strictly procedural knowledge to process knowledge: information about what is happening to the objects as they are being acted upon in each step.

The demonstration performance introduces the steps and their sequences. All procedural information is presented to the learner the name of the step, its order in the sequence, the identity of the person responsible for performing the step, any tools used in performing the step, and a mediated demonstration of the step being performed. The only performance required of the learner is to repeat the performance of the step immediately following the demonstration. Process knowledge is not instructed.

The practice levels slowly fade the cues and increase the requirements for the performance. Additionally, process knowledge is taught. The learner may be asked to predict the value which a property of one of the objects in the environment will have after the step is performed. The learner may also be shown a summary of the effects of a step on all properties simulated by the instruction. From this summary, the learner may request an explanation of the value of any property, or explicit instruction linking sets of input values to output values for a selected property.

Self-check withdraws all cues and requires full performance of the procedure by the learner, including proper ordering of steps, selection of tools, and performance of the steps.

Each performance of the procedure may include an introductory and conclusion media presentation. In addition, at all times that an explicit learner response is not prompted, the learner may explore any object that is visualized or referenced on the screen. This exploration may lead to the display of information, a media presentation, or explicit instruction. For example, clicking on a device brings up a dialog which describes the device and lists its properties and their states.

Step performance is via direct manipulation of a visualization of the performance environment. Each object in the environment (for example, devices, device controls, device indicators) is represented visually by a graphic. Objects which always appear the same are represented by a single graphic. Objects whose appearance varies with their state are represented by a set of graphics, to each of which the author attaches a rule indicating for which state values that graphic is valid The appropriate graphic is automatically selected for display during instruction based upon the state of the represented object. Objects which represent controls (buttons, switches, knobs, dials, etc.) respond during a learner performance with a popup menu of valid settings when clicked by the learner. The learner then selects a setting. During a demonstration performance, the algorithm selects the setting automatically.

Whenever any action occurs that changes the state of a control, the simulation which underlies the visualization is updated. Each object may have defined upon it one or more properties which represent the state values of the object. For each property the author defines a set of rules which determine the value of that property. During a simulation cycle the rules of all properties are interpreted and the property values updated. Immediately afterward, the rules for all visualizations are re-examined based on the new property values, and the visualizations updated. From the learner's point of view, he or she clicks on a control and selects a setting; the control then changes to represent the new setting (for example, a toggle switch goes from off to on) and possibly other devices change as well (for example, a light may be illuminated).

The simulation is not free play -- the learner may perform any action but only correct actions are simulated Incorrect actions receive feedback, which may range from a message indicating that the action is not correct to a demonstration of the correct action. The type of feedback is determined based on the performance level and the learner's prior actions. The decision to not support free-play simulation is based primarily on the goal of authoring efficiency. It turns out that the definition of the rules which govern the simulation is the most difficult and least familiar task authors must perform. A free-play simulation requires a rule set considerably more complete and complex than is the case for path-based simulation. A functional and instructionally useful path-based simulation may be constructed with a small set of rules and may be defined with a relatively limited knowledge of the actual inner workings of the device being simulated

It should be mentioned that an instructional simulation of the type described here is built for different purposes and under different constraints than other simulations. Many simulations, for example those used in engineering design or prediction of natural phenomena, are used to study the system being simulated for the purpose of acquiring new knowledge. These systems are only useful to the extent that they fully and accurately capture the dynamics of the system. An instructional simulation of the type described here is designed to support a reactive environment for practicing the performance of a procedure. It need only account for existing knowledge and not for new or unknown system attributes. The conditions of its use are constrained. It need not completely model the system, only that part of the system which is instructionally relevant to the objective at hand. Nor must the model be fully accurate, in fact, it may be desirable to implement a model of system function that is simplified as that may be more appropriate for a particular category of learner. For example, the appropriate model for an operator of a device is not the same as the more detailed model necessary for repair or design of the device.

Authoring

Authoring involves two activities: analysis and resource preparation.

Analysis

The analysis required is more involved than is the case with existing authoring systems. The author is required to carefully describe both the performance and the environment of the performance. A set of pre-defined generalized object descriptors, or classes, is provided. These are of three types: content, instructional, and visualization objects. Content objects represent subject matter: examples are procedures, devices, controls, and personnel. Instructional objects define how the instruction is to be carried out; examples include lessons, performances, and guidance settings. A set of pre-defined instructional objects is provided so that authors need not be concerned with these, however for those authors with instructional design expertise the instructional objects are available for customization. Visualization objects link to the resources (graphics, video, audio) which represent the objects of the instructional environment.

Each class defines a set of slots, or attributes, that are valid for objects of that class. Authoring consists of creating new objects (instances) based upon the patterns stored in the classes and providing values for the attributes. Attribute values may be primitive types such as strings or numbers, complex types such as rules, or links to other objects. For example, the step object has attributes for the action to be performed (a complex type which uses a simple grammar to specify a setting for a control, for example "SET main power switch TO on"), links to the person who performs the step and the tools used in the step (links to other content objects), and a link to a visualization object demonstrating the step being performed.

Though the analysis required exceeds that normally performed, there are two important advantages. First, it makes possible the entire approach of re-usable algorithms. The algorithm is defined based upon the object classes, and not upon any specific subject matter. Thus the algorithm is designed to teach a step, its personnel, its tool, its demonstration and action without knowledge of any specific step, personnel, tool, etc. During instruction, the specific elements are retrieved from the knowledge base and substituted. The algorithm may be re-used many times, with different subject matter.

Second, the instruction is generated directly from the analysis. This differs from the standard approach of a sequence of translations, from analysis to design to final development in some programming language. This has several important consequences. With a single representation of the content there are no problems of inconsistent representations: the analysis is always up to date and reflects exactly what is taught. Changes at the analysis level immediately update the instruction. As a result, life-cycle costs are dramatically reduced. Updates and modifications to existing lessons may be carried out at the analysis level, which is the most accessible for humans (as compared to the computer code). Within the knowledge base, each element is represented exactly once, no matter how many times it is used in the instruction. Thus updates to a knowledge object may be made with full assurance that all uses of that object have been updated

Another important consequence is support for rapid prototyping. Not all attributes need be assigned values in order for instruction to run. In fact, the algorithm is designed to function completely with only a few attributes defined, and even finished instruction may leave many attributes undefined Missing data is replaced by defaults and place holders. This has the most important effects in the instructional and visualization objects. All instructional objects have pre-defined default values, so that no authoring need be done at all. Visualization objects will provide place holders, so that the instruction may be run with full user interaction prior to the development of all or any graphic or media resources.

Rapid prototyping is an important element of the approach towards reducing overall development costs Developing prototype instruction very early in the analysis phase means that alternative approaches may be explored with feedback from colleagues and other interested parties. Because the prototype is the instruction, rather than an abstract representation of the potential instruction (for example, a list of objectives or a storyboard) it is more easily interpreted by those with less expertise in the representations used by instructional developers for example, management, clients and potential users Changes suggested by these people may be incorporated at far lower cost than is the case when the first working version of the instruction is ready only when the development schedule is 90% complete.

Resource Preparation

Resource preparation is the development of graphics, video, audio, and any other media to be included in the instruction. The basic work of resource preparation is unchanged: graphic artists must still draw pictures, video production staff must still shoot video.

While the re-usable algorithm approach cannot automate picture drawing, it can positively affect the overall cost of resource development.

First, unlike much current video-based instruction, the resources do not carry the principal instructional message. The algorithm accomplishes this. Resources primarily visualize the environment and supplement the principal message. Fewer overall resources may be needed.

Resources may be re-used. Often, a scene is built from a series of graphics, each of which may be re-used in a variety of situations. Life-cycle costs are reduced when a part of a device changes, as only that part need be re-shot rather than the entire scene in which that part appears.

The rapid prototyping, which may precede resource development, or may use cheap resources such as camcorder video or audio captured by the developer, means that the instruction may be more completely developed and critiqued before resource development commences. This should reduce the need to re-do media production due to design changes.

Finally, the management of the overall development process will have more information available that can be used to streamline media production. Reports may be generated from the knowledge base indicating the number and types of media resources specified in the analysis. This information may be used to coordinate and schedule media production.

Prior Work

This research extends our earlier work on transaction shells (Li & Merrill, 1990; Merrill, Li, & Jones, 1991). This prior work focused on teaching part structure and did not incorporate simulation. The knowledge representation builds on that previously reported in (Jones, Li & Merrill, 1990), which in turn was influenced by work in semantic data bases (Hull & King, 1987; Peckham & Maryanski, 1988) and object oriented programming (Agha, 1987, Goldberg & Robson, 1983). The simulation aspect is influenced by (Towne & Munro, 1988) and (Halff, 1990). The instructional strategy has roots in (Merrill, 1983) and (Gagne, 1985). Notions about mental models are widely published, see for example (White & Frederiksen, 1990). Work on rapid prototyping in software development has some parallels to courseware development, see (IEEE, 1989).

Further Work

This project has completed the first of three phases. In phase 2, a number of extensions and modifications are planned. The most important of these are:

- extensions to the content objects to represent conditional and repeated steps, branches, waits
- instruction of the materials used in a step
- specialized instruction for steps flagged as critical or dangerous
- specialized feedback for actions flagged as common or dangerous errors
- support for all MCI devices
- support for procedural attachments to resources so that they may update themselves
- incorporating structural instruction: device parts and connectivity
- support for multiple environment locations and explicit instruction for finding a control in a location
- maintenance of individual student profiles and customization based on the profile across sessions

We envision eventually creating a library of such re-usable algorithms, of which device operation procedures are just one example.

References

Agha, G. A. (1987). *ACTORS: a model of concurrent computation in distributed systems.* Cambridge, MA: MIT Press.

Gagne, R. M. (1985). *The conditions of learning.* New York: CBS College Publishing.

Goldberg, A. & Robson, D. (1983). *Smalltalk-80: the language and its implementation.* Reading, MA: Addison-Wesley.

Halff, H. (1990). *Automating maintenance training.* Arlington, VA: Halff Resources.

Hull, R., & King, R. (1987). Semantic database modeling: survey, applications, and research issues. *ACM Computing Surveys* 19 (3), 201-60.

IEEE (1989). Rapid prototyping in software development. Special issue of *Computer* 22 (5).

Jones, M. K., Li, Z., & Merrill, M. D. (1990). Domain knowledge representation for instructional analysis. *Educational Technology* 30 (10), 7-32.

Li, Z. & Merrill, M. D. (1990). Transaction shells: a new approach to courseware authoring. *Journal of Research on Computing in Education*, 23 (1), 72-86.

Merrill, M. D., Li, Z., & Jones, M. K. (1991). Instructional transaction theory. *Educational Technology* 31 (6), 7-12.

Merrill, M. D. (1983). Component display theory. In C. M. Riegeluth (Ed.), *Instructional Design Theories and Models.* Hillsdale, NJ: Lawrence Erlbaum.

Peckham, J. & Maryanski, F. (1988). Semantic data models. *ACM Computing Surveys* 20 (3), 153-89.

Towne, D. M. & Munro, A. (1988). The intelligent maintenance training system. In . Psotka, D. Massey, & S.A. Mutter (Eds.), *Intelligent Tutoring Systems: Lessons Learned.* Hillsdale, NJ: Lawrence Erlbaum.

White, B.Y. & Frederiksen, J.R. (1990). Causal model progressions as a foundation for intelligent learning environments. *Artificial Intelligence* 42, 99-157.

Acknowledgments

This is a team effort and would not be possible without the efforts of a number of people, including M. David Merrill, Zhongmin Li, Scott Schwab, Mark Lacy, Leston Drake, Vicki Napper, Jean Pratt, Richard Cline, and Thor Anderson.

We especially appreciate the support of our sponsors, the Training Systems Program Office of the U.S. Air Force Aeronautical Systems Center.

IconiCase: A visual system for rapid case review

M. KALTENBACH
Department of Management and Information Sciences
Bishop's University, Lennoxville, Quebec, J1M-1Z7, CANADA
E-Mail: kaltenba@iro.umontreal.ca

B. PREISS
Département de Biochimie
Faculté de Médecine, Université de Sherbrooke, Sherbrooke, Québec, J1H-5N4, CANADA

Abstract: A computer system is proposed to assist humans in the task of classifying and memorizing medical cases with the help of assemblies of icons, called Concept Graphics (CG). This leads to a form of Electronic Document Delivery in which information is accessed at various levels of details and the transitions between the levels is gradual. Also it is argued that rapid review of large sets of documents for content and differential analysis is a valuable activity that can be effectively supported by electronic systems.

Since Antiquity Medical knowledge has essentially been case-based, despite the relatively recent growth of knowledge that links biological facts and processes into increasingly complex and accurate causal models. Medical students and professionals still use reference manuals, such as The Merck Manual (Merck, 1992) describing large numbers of distinct pathologies. Apart from very specific and narrowly circumscribed domains, applications of Expert Systems so far have not provided a totally satisfactory answer to the problem of automating or assisting diagnostic reasoning. Physicians remain reluctant to adopt computer decisions made on a narrow information base. Consequently the more recent systems are oriented toward combining intelligently assisted information retrieval and decision making (Kushniruk, 1994). Apart from providing a richer context for decision making, this fulfills the additional role of helping humans in forming useful categorizations and memorizing large numbers of medical pathologies (Hirtle, 1986).

In this paper we describe a computer system, currently under development and called IconiCase, to assist humans in the task of classifying and memorizing medical cases. More specifically, the system exploits assemblies of icons, called Concept Graphics (CG) as pointers to medical cases (Preiss 1992). This leads to a form of Electronic Document Delivery in which information is accessed at various levels of details and the transitions between the levels is more gradual than in outline systems such as Microsoft's More (Symantec 1992) or window oriented hypertext systems such as NoteCards (Halaz, 1987) or Intermedia (Yankelevitch 1985). In (Kaltenbach, 1988-1991) various ways of providing smooth transitions in information delivery are explored. It is argued that even short gaps in the way information is presented to humans, such as when details are added in separate windows to offer a complement information to a particular point in a mathematical proof, lead to cognitive difficulties, that could be prevented by better human/machine interfaces. Another important aspect of the system, is the active role given to users in organizing their access to the information.

Access to information by the intermediary of icons has received much attention since the beginning of computer graphic interfaces. However in most of the applications proposed so far the meaning and use of icons is quite restricted. Desktop icons for operating systems interfaces have a very simple meaning that allows little variations, if at all. There are very few systems that place meaning in the composition of icons. Systems for visual programming (TGS, 1993) only provide a more readable syntax for programs; the iconic compositions do not have to be interpreted as sets. Iconic compositions, used to formulate queries on a database (Erradi, 1988),(Chang, 1990) form the closest work to what we are proposing; however only a partial use of icons is made to access the documents. The retrieved documents themselves are not assessed with the help of iconic compositions. In addition in the early applications the need to translate queries in a database language, such as SQL, placed severe constraints on the kind of information that can be conveyed through iconic compositions. In this work we attempt to relax the constraints on the kind of information that can be encapsulated into iconic compositions and seek computer solutions to attenuate if not remove completely the ensuing cognitive difficulties associated with the use of icons. In the first part of this paper we define Concept Graphics and propose them as useful intermediaries between text and human mental representations.

Concept Graphics

In the following the detailed descriptions of medical pathologies that form a corpus of documents are called cases. Cases are obtained in the form of texts that can be very detailed, such as entries in The Merck Manual. A case has a unique textual identifier plus textual and iconic indexing terms. In addition it is represented by a composition of elementary icons, called a Concept Graphics (CG), that captures the main characteristics of the case, such as symptoms, treatments, factors such as age, habits, etc. Thus the CG components need not be homogeneous. The choice of these characteristics may reflect the biases of a particular expert, or general user of the system. An example of a CG corresponding to the case "acute pancreatitis" is given in Figure 1 . Note that the use of color make the CGs more easily identifiable..

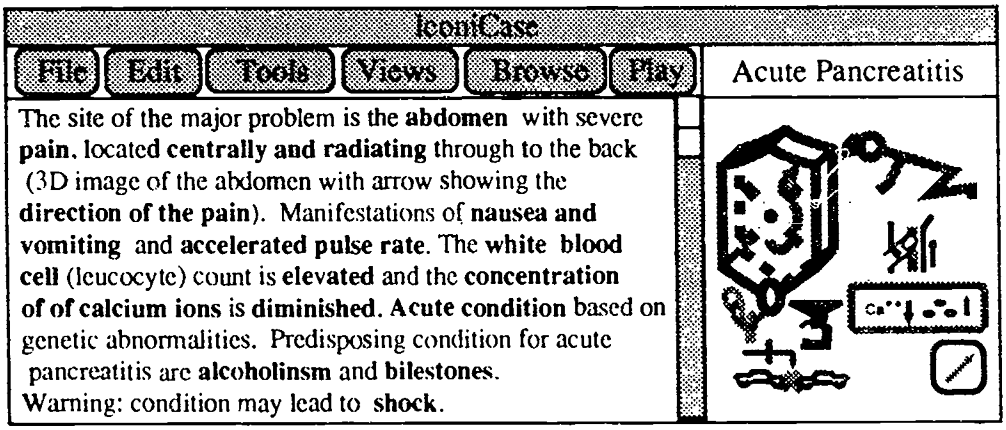


Figure 1. A concept graphic and associated short text

For us CG are mnemonic devices, akin to the chimerical compositions used by medieval orators to summarize the main points of the sermon they were going to make. For a CG to be a mnemonic device means that its meaning is not just in the icons themselves, but that it is remembered by the viewer on the basis of cues offered by the icons. It should be noted that the early forms of Mesopotamia writing were assemblies of pictograms representing concrete things or actions (Bottero, 1993). It is only later that syllabic meaning was ascribed to pictograms which, by becoming more abstract, led to the alphabet.

As much as possible icons are designed to be interpreted without the help of textual explanations; e.g. a broken bone, a brain, etc. However there is a limit to the complexity of meaning that can be expressed by icons in the form of familiar concrete objects and abstract icons are a necessity. Our approach to resolving the problem of understanding unfamiliar icons is to link the individual icons as well are icon sequences to text and to introduces the use of new icons in a progressive manner to users. This means that what users are shown in a CG depends on what icons they understand. This also implies that a user model (Sleeman, 1984) be maintained by the system. When a new icon is introduced its textual interpretation is automatically provided and its correspondence to the text of the case is highlighted. In Figure 1, the bolded letter words in the left text field correspond to the icons in the CG on the right. Later we provide additional details on how practically users are made familiar with icons and iconic compositions, as well as the kind of assistance they can have when "reading" iconic compositions.

Discounting for the time being the familiarity/unfamiliarity aspect, the advantage of using CG to access information is the exploitation the human visual processing abilities adapted to deal with objects in multiple dimensions such as space, shape, color, texture . There are several apparent consequences:

- For the same speed of access, more information is obtained by the user than if it had been conveyed in textual form; or conversely the same amount of information is obtained more rapidly. Speed of access is an important factor when dealing with large amounts of documents and traditional access methods through Boolean queries or natural language processing are too ineffective or cumbersome. Users may simply not have the time available for a long survey. To bring back to mind the content of a large set of documents a sort of rapid browsing called "flying through hypertext" (Lai, Manber, 1991) is needed.

- the many dimensions of iconic compositions and the ability to scan through many cases in a short time result in ways of classifying cases that would not have been obtained through statistical classification methods (Preiss, 1992) because when using these methods often one must know what to look for, either ahead of applying the method or in the interpretation of the results. Machine Learning algorithms (Kodratoff, 1988) and other discovery methods in databases, also fall under that criticism.
- it may be the case that the composition of a CG conveys more meaning than just the sum of its parts.

The limitations of using iconic compositions have been extensively studied and reported in the literature (Iwam, 1991). Experiments comparing the use of abstract icons and text menus have concluded with no advantage for icons (Benbasat, Todd, 1993). Thus,

- interpreting individual icons can be difficult
- it is not clear what meaning to ascribe to a juxtaposition of several icons
- icons use a lot of valuable screen space.
- icons and iconic compositions are cumbersome to manipulate.

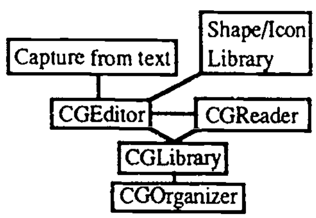


Figure 2. Main modules of Iconicase

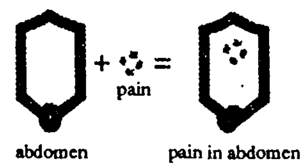


Figure 3. Iconic composition

The system IconiCase is an attempt to capitalize on the advantages of using Concept Graphics, while striving to iron out the difficulties. Figure 2 presents the main modules of the system IconiCase to create and exploit Concept Graphics for surveying and analyzing large sets of cases (i.e. documents). The modules are organized into two main groups corresponding to authoring/editing of CGs and of exploiting CGs in a consultation mode.

Creating Concept Graphics

Capture from Text

Source text, such as entries in The Merck Manual, must be summarized in order to regroup the main features of a case and provide a textual explanation that closely match the corresponding CG. When completed this module will offer a more efficient way of creating the summary text from the original text than by standard cut and paste operations; a user selection of words, group of words and sentences, will be sent to the summary window with only one keyboard operation. Also the system will automatically create links between corresponding parts of the original and summary text, making possible the simultaneous highlighting of corresponding parts of text.

The icon editor

Icons are grouped set of basic shapes that form a sort of collage. This means that basic shapes are not restricted to be rectangles and that they include regions that are transparent. Figure 3 shows the basic shapes "abdomen" and "radiating pain" and the composite icon "pain radiating from center of abdomen". Icons are organized in a multiple inheritance object hierarchy of basic shapes. Each shape or icon has a textual meaning and is indexed by keywords.

New icons can be made by screen captures of basic shapes created with a standard painting or drawing program. There is a tool for defining particular regions of a captured image as transparent or opaque with a color the user can choose. Regrouping basic shapes, naming and indexing an icon is done in a standard way. It is also possible to create new icons by modifying existing ones retrieved through an icon browser or from existing Concept Graphics.

The Concept Graphic editor

Concept Graphics are assembled manually but we are planning to introduce some automated assistance in selecting the icons and assembling the CGs. For instance the keywords recognized form the text of a case can be automatically translated into icons appearing in the right field of Figure 1. Use of a hierarchical thesaurus (Côté 82) is also considered, to select icons higher in the hierarchy when a recognizable medical term in the text does not have yet a corresponding icon.

The relative placement of icons in a Concept Graphic can be (at least partially) automated by default or production rules. Default placement rules could be statistical. For instance place the new icon at the average position of that icon in already created CGs. Production rules would supersede that default placement to locate semantically related icons at the same relative place in a CG.

Though icons appear with irregular contours, they are included in defining rectangles. This makes it possible to apply a tool we have developed (Kaltenbach, 1991) to compact CGs in as little screen space as possible, while seeking to preserve the relative positions of the composing icons and avoiding the overlap of icons.

It is also in the CG editor that the correspondence between text and icons is established. It has the form of links that enable the simultaneous highlighting (by flashing) of corresponding parts of text and icon groups when a selection in the left text or right image field of Figure 1 is made by the user. A default selection of text or graphic objects is obtained by simply positioning the mouse cursor over the object. Other selection modes for more detailed or coarser selections are also available.

Once a CG is edited it is placed in a CG browser under an appropriate category.

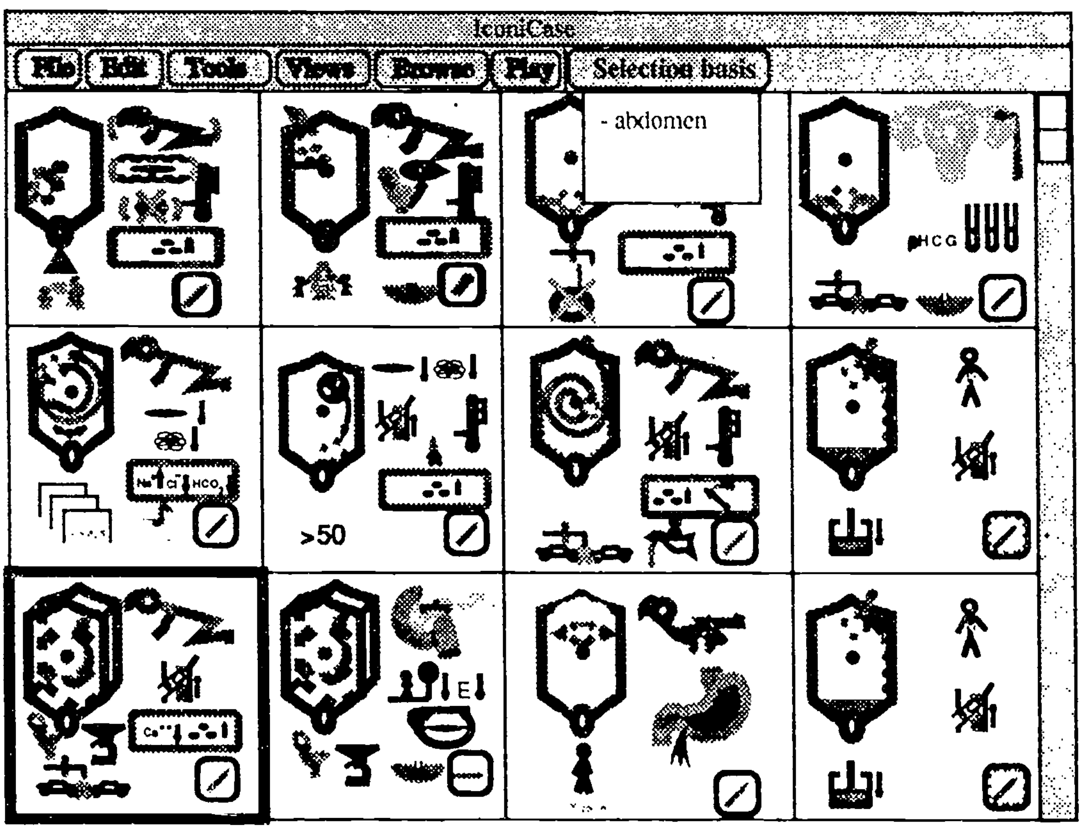


Figure 4. Viewing cases sharing attribute "abdomen" with the case "acute pancreatitis"

Accessing concept Graphics

In addition to helping users understand a CG, IconiCase provides alternate ways of efficiently scanning through large collections of Concept Graphics. Current implementation includes a CG player, a textual find utility, a browser by semantic categories (e.g. heart diseases, pulmonary diseases,..).

the CG reader is a facility that automatically highlights the keywords, sentences in the left field of Figure 1, together with corresponding icons in the right field. The speed of scanning can be controlled by the reader.

the CG finder is a facility that enables the user to select particular icons or icon components on the right of Figure 1, and obtain the text list or collection of CGs that share the selected features. Figure 4 gives the result of a search when the icon shape "abdomen" is selected. A zoom facility for the CGs in Figure 4 is projected.

the CG player enables users to view CGs (Figure 1) in succession at a speed they can control. The sequences of CGs that can be viewed are defined using the CG sequencer.

the CG sequencer currently permits the definition of CG sequences that correspond to:

- user defined sequences by associating markers on CG seen as in Figure 1 or on clicking on CG as seen in Figure 4.
- sequences corresponding to a semantic category in the CG browser
- finally it is projected to cluster cases on the basis of more semantically related definitions of distance between cases. This may rely only in part on a measure of distance between individual icons.

Any defined sequence can be reordered according to a measure generality for CGs. A detailed definition of this measure would make this paper too long, so we provide only the underlying idea. A partition of the set of icons (and icon components) available in the system is created under the equivalence relation that all the elements of a subset are related if they all index a set of cases that is maximal in the sense that no super set of cases is indexed by all these elements. The number of cases associated with an element of the partition of icons gives a measure of the generality of that icon. Then the generality of a case is obtained as the average generality of its component icons.

The sets of cases as just defined can be ordered by inclusion and thus be accessed as the nodes of a graph. A particular node is seen in our system as a sequence of CGs (Fig. 1) or as a two dimensional display (Fig. 4). The user can traverse links in that graph to refine or expand the generality of the component cases entering in a sequence, while focusing on a set of icons of current interest. In fact there are many ways in which significant navigation can be achieved in that graph and this will be the object of subsequent paper.

Using IconiCase

First we return to the question of how users get to understand individual icons and iconic compositions. At one extreme a user may exploit an already processed collection of cases. By this we mean that the cases have already been summarized and associated with CGs by someone else, acting as author. The user may then choose either to get explanations of complete CGs or to be shown no more than i new icons (i=1,2,...) per CG. By pointing to an individual icon or by selecting a component the user gets the correspondence with the text (left field in Fig. 1) and a more formal definition of the icon in the text field below the CG in Fig.1. Explanations are also available for groups of icons when they are selected by the user.

The other extreme is to let the users be also the authors of the CGs. The interpretation of the icons and groups of icons then does not poses problems; user made associations are in the essence of mnemonic approaches (Yate, 1966). In counterpart the practical value of the system IconiCase then hinges critically on how easy it is to create CGs. We have described some early steps in that direction.

In this paper we are presenting CGs as a prop to memory. For instance it could be used by students in order to assist them in reviewing for exams.

There is another objective to the system IconiCase Note that when viewing a large number of CG, it may not be necessary to understand the individual icons in order to make some inferences. For instance some overall pattern may be detected viewing a large collection of CGs leading to the identification of causal relationships (Knowledge base rules). The scanning of CGs could also be viewed as a complement to automated knowledge discovery approaches in databases. The rapid human scanning of cases obtained by the automated approaches would further validate or invalidate the newly extracted rule. This is important because usually many rules are discovered by the automated discovery approaches and screening through them to get the really meaningful is a very tedious task.

Future work and conclusion

As an interface for the delivery of large corpora of documents, a system like IconiCase requires many adaptive features and much fine tuning to gain acceptance with a wide variety of users. We have explored some of these dimensions but much remains to be done. We plan to perform experiments with users to study the time needed to learn the meaning of icons and of groups of icons. Also we shall study possible tradeoffs between the depth of details and speed the CGs can be displayed to humans. The criterion of evaluation should how correctly cases are reminisced.

The approach is of course not strictly limited to medical cases. Also it could be considered as an extension of and a complement to case-based reasoning systems (Hammond, 89).

References

Benbasat, I.; Todd, P. (1993). An experimental investigation of interface design alternatives: icon vs. text and direct manipulation vs.menus. *International Journal of Man-Machine Studies,* Vol: 38 Iss: 3 pp. 369-402

Berkow Robert, (1992). The Merck Manual of Diagnosis and Therapy, *Merck & Co., inc.*

Bottero, Jean, (1993). La naissance de l'écriture, *Historia,* No 24, pp55-59, July-August.

Chang, Shi-Kuo, (1990). A visual Language Compiler for Information Retrieval by Visual Reasoning. *IEEE Transactions on software Engineering.* Vol 16, No 10. October.

Côté R.A. (ed) (1982). Systematized Nomenclature of Medicine. Skokie, Illinois. *College of American Pathologists.*

Erradi, M., Frasson, C. (1988). Visual Interaction using an Iconic System. The Visual Computer, *International Journal of Computer Graphic,* vol.4, n2, Springer-Verlag, pp98-108.

Halasz F. and Trigg R., Notecards in a Nutshell, (1987). *CHI & GI'87 Conference Proceedings,* Toronto, Canada.

Hammond, Kristian J.(1989). CASE-Based Planning, *Academic Press.*

Hirtle, S.C. and Mascolo, M. F.,(1986). Effect of Semantic Clustering on the Memory of Spatial Locations, *Journal of Experimental Psychology: Learning, Memory and Cognition,* Vol.12 No.2, p182-189.

Iwama, M. (1991). The comprehensive faculty of icons-drawings, meanings environments , *Review of the Communications Research Laboratory* (Japan) . Vol: 37 Iss: 4 p. 479-95

Kaltenbach M., (1991). Interactive Classification Analysis via a set-to-set browser. (concise paper), 2nd ASIS *SIG/CR Workshop on Classification Research,*Washington, D.C.

Kaltenbach M., Robillard F., Frasson C., (1991). Screen Management in Hypertext Systems with Rubber Sheet Layouts. *Proceedings of Hypertext'91 conference,* San Antonio, December p15-18.

Kaltenbach M. and Frasson C., Dynaboard: User Animated Display of Deductive Proofs in Mathematics. *International Journal of Man-Machine Studies,* 30.

Kodratoff, Yves, (1988). Introduction to Machine Learning, *Morgan Kaufmann Publishers.*

Kushniruk Andre, (1994). Submitted for presentation at the conference *ED-Media,* Vancouver, Canada

Lai, Patrick and Manber, Udi. (1992). Flying through Hypertext, *Proceedings of Hypertext'91 conference,* San Antonio, December pp123-132.

Preiss, B., Kaltenbach, M., Zanazaka, J., Echave, V. (1992). Concept Graphics: A language for Medical Knowledge. *Proceedings of Annual Symposium on Computer Applications in Medical Care (SCAMC),* Baltimore. November.

Sleeman, D.H. (1984). A user-modelling front-end subsystem. *International Journal of Man-Machine Studies,* vol. 23, no 1, pp71-88.

(Syma 92) More 3.0, (1992). Symantec Corporation, Cupertino, California.

TGS Corporation (1993). Prograph 2.5.

Yeats, Frances A. (1966). The Art of Menory, *Chicago University Press.*

Children's Independent Exploration of a Natural Phenomenon by Using a Pictorial Computer-Based Simulation

MARJATTA KANGASSALO
University of Tampere, Department of Education
P.O.Box 607, FIN-33101 Tampere, Finland
Email: kamaka@uta.fi

Abstract: This paper describes an investigation examining the constructing of a conceptual model of a selected natural phenomenon by children when using a pictorial computer simulation of that phenomenon. The paper concentrates on describing children's exploration process in relation to their conceptual models before and after the use of the simulation. The selected natural phenomenon was the variations of sunlight and heat of the sun as experienced on the earth related to the positions of the earth and the sun in space. Children's conceptual models before the use of the simulation formed a starting point from which the exploration of the phenomenon was activated. Children's exploration seemed to contain, for example, wandering here and there, investigating and seeking for something and experimenting with aim. During the exploration process occured the construction of conceptual models in varying amounts but the construction seemed to follow a correctly directed conceptual model of that phenomenon.

This paper is one part of an investigation whose aim was to investigate to what extent the independent use of a pictorial computer simulation of a selected natural phenomenon could be of help in the organizing of the phenomenon and the forming of an integrated picture of that phenomenon. Attention was paid to the constructing of a conceptual model of the phenomenon by children and children's exploring process during the use of the computer simulation. This paper concentrates on examining children's exploring process and its relation to the constructing of children's conceptual models by using a pictorial computer simulation of that phenomenon. The children taking part were seven-year old Finnish first-graders. They had at school no formal instruction before or during their exploration process of this natural phenomenon. The autho: has not found any corresponding works.

In this work a conceptual model of a natural phenomenon is seen as a mental construct which is based on activated information stored in the interconnected neural networks of the brain concerning a phenomenon in question. A conceptual model has been formed from the basis of the regularities of a natural phenomenon. It covers the events, objects, properties and relations of a natural phenomenon. A conceptual model makes it possible to think about a phenomenon, to describe it, and to explain and to predict the events of the phenomenon. In a conceptual model, conceptual refers to activated information in the neural networks, and a model of a phenomenon forms when interconnected neural networks concerning a phenomenon are activated.

This paper first describes the pictorial computer simulation of a selected natural phenomenon. After it procedures for the examination of children's exploration process are described and finally children's exploration process and its relation to changes in children's conceptual models before and after the use of the simulation is examined.

Pictorial Computer Simulation

The selected natural phenomenon for the pictorial computer simulation was the variations of sunlight and heat of the sun as experienced on the earth related to the positions of the earth and the sun in space. In the simulation it is possible to explore the variations of sunlight and heat of the sun and their effects on the earth in a natural environment and the origin of these phenomena from the ba~is of the interconnections and positions of the earth and the sun in space. The simulation concentrates on phenomena which are close to the everyday experiences of children, such as day and night, seasons, changes in the life of plants and birds etc. The simulation program has been implemented in such a way that the knowledge structure and theory of the phenomenon are based on events appearing together with the phenomenon in question, and these events are illustrated. In the simulation all events and necessary elements are represented as pictures and familiar symbols. At the earth level the pictorial simulation represents the surrounding world, its phenomena and objects in a very natural and realistic way. In exploring the phenomenon at the space level the interrelations of the earth and the sun are represented with the help of an analogue model. The selected place on the earth from where the phenomenon has been modeled and simulated for a computer is the suburb of Lentävänniemi in Tampere, in Finland. The children who participated in this research live in this area. Therefore the exploration of the phenomenon with the pictorial simulation takes place in an environment familiar to the children.

The computer program begins so that on the screen appears the earth revolving on its elliptical orbit around the sun. A child can see the whole revolution of the earth around the sun and can continue at any time to the next phase where the name and age of a child is asked. Next a map of Finland appears on the screen and by pressing the position of the city of Tampere on the map a view from Lentävänniemi toward the south is seen on the screen.

The exploration of the phenomenon starts in Lentävänniemi on the 1st January at midnight. On a big screen is seen at that moment a dark, snowy, winter landscape in a direction of the south. The exploring can be continued by using icon pictures under the picture of the landscape, for example, a clock, a pictorial calendar, a picture map with different points of the compass and a space shuttle. The exploring of the phenomenon can be continued in many ways hour by hour, day by day, month by month, at different points of the compass and so on. At any moment it is possible to take a space shuttle and to look at the interrelations and positions of the earth and the sun at the space level. At the space level it is also possible to explore the interrelations and positions of the earth and the sun by using icon pictures which show the time in the same way as at the earth level. At every moment it is possible to choose how to continue exploring. Exploring can proceed at every moment using the existing alternatives.

When exploring at the space level the earth, rotating around its axis and revolving around the sun on its elliptic orbit, appears on the screen. The plane of the earth's orbit is shown as viewed from directly above this plane. When exploring the phenomenon at the earth level the following may appear on the screen: the changes of darkness and lightness daily with an accuracy of one hour, the sun's positions in the sky every hour, the place and time for the sunrise and the sunset every day round a year, typical plants, birds and animals according to seasons and so on. With the help of the icon pictures, binoculars, magnifying glass and microscope it is possible to explore flowers, trees, leaves, insects, birds, spores, animals and so on in more detail. On the screen a selected object is seen as bigger. At the space level with the help of a telescope it is possible to look at the earth at a larger size so that, for example, Finland is seen more clearly. From this picture of the earth it is possible to continue further and to see the map of Finland, and finally an air photo of Tampere. In the simulation it is also possible to explore the mutual size of the earth and the sun and to get an image of the distance between them. Also the position of the earth in the whole solar system can be seen on the screen. The pictorial computer simulation is constructed so that it is very easy to use and it does not presuppose an ability to read or write. A pictorial computer simulation (Kangassalo 1991) is described in details in the article by Kangassalo (1992).

Examination of Children's Exploration Process and Description Techniques

When a child explored a phenomenon by using simulation each press of the mouse was recorded in the computer's memory. Presses were recorded so that it was possible afterwards to recall, from the computer's memory the children's exploration pathways and to see each child's exploration pathway on the screen. On the screen a child's exploration pathway can be seen press by press as pictures like it occured in the exploration situation. Children's exploration pathways were recorded from the computer's memory on to paper and a description technique was developed for them. The description technique makes it possible to follow children's exploration process as it happened in the original situation. In the description technique it can be seen on which levels of the phenomenon a child's exploration took place. Table 1 shows this technique. In the technique the following levels can be separated: earth/space, the point of the compass on the earth, month (including season and month), day (including number of a day, time, awake/sleep), research mediums (at space level also the scale and the solar system) and finally details. In Table 2 an example is shown about a child's first exploration pathway by using the described technique.

Table 1.
The description technique for children's exploration pathway.

<u>1 st January 00 o'clock on the earth the point of the compass is south</u>

- point of the compass
  - season
  - month
    - number of a day
    - time
    - awake/sleep
      - research medium
        - detail

space shuttle to space:earth-sun

- season
- month
  - number of a day
  - time
  - awake/sleep
    - research medium
      - map of Finland
      - air photo of Tampere
    - scale
    - solar-system
    - scale

back to earth-sun

space shuttle to earth

Table 2.
The first exploration pathway of a child.

<u>1 st January 00 o'clock on the earth the point of the compass is south</u>

- awake
- east
  - season
    - 2 nd
      - binocular
        - sparrow

space shuttle to space: earth - sun

- telescope
- scale
- solar system
- scale

back to earth - sun

space shuttle to earth

- season
  - magnifying glass
    - raspberry flower
    - measuring worm
    - ladybird
    - dragonfly

space shuttle to space: earth - sun

- scale

back to earth - sun

- 12
- awake
  - telescope
    - map of Finland
    - air photo of Tampere

space shuttle to earth

The second description technique has been developed with the purpose of summarising and clarifying a child's exploration subjects at different exploration times. An example of this technique is seen in Table 3. In this table has been described all icons by which it is possible to explore the phenomenon both at the earth level and at the space level. In this table can be seen what a child explores, how many times and how extensive an exploration is. By examining all a child's exploration times as separate tables it is possible also to see in which order a child explores the things of the phenomenon. Both description techniques support and are complementary to each other. The descriptions of the exploration pathways shows a child's exploration processes step by step and from tables they can be seen in summary. By means of the descriptions of exploration pathways and tables can be followed the construction process of children's conceptual models.

Table 3.
The first exploration pathway of a child as a table.

Child: 1
Date: 20.11.1991
Number of frames: 28
Time: 11 min

	On the earth				In space
	south	east	west	north	
season		2			
month					
day		1			
hour					1
awake/sleep	1				1
binoculars		1			
details		i			
magnifying		1			
details		4			
microscope					
details					
telescope					2
map					1
photo					1
scale					3
solar system					1
point of	1	shift in space			2
space shuttle	4				

Children's Exploration Process and Its Relation to Children's Conceptual Models

Before and after the exploration of the natural phenomenon children's conceptual models have been elicitated by using procedures for this aim (see Kangassalo 1993a). Before the use of the simulation children's conceptual models were at very different levels. Some children's conceptual models of the phenomenon were quite undeveloped, and some others' very developed. Only some children's conceptual models contained a clear misconception. During the use of the simulation in children's conceptual models changes occured. The most significant change was that the interconnections of different phenomena and entities began to be constructed and the construction seemed to follow the currently accepted scientific knowledge.

In the construction of the interconnections of phenomena and entities the following phases could be seen: the existence of interconnections had been discovered, the organization of interconnections took shape (order, continuity and regularities), the amount of interconnections increased, the reorganization of interconnections took shape: interconnections ceased to exist or/and interconnections replaced by other interconnections.

In the natural phenomenon in question the main direction of constructing by children tends to be as follows:

1. The earth begins to revolve around the sun.
2. Seasons are organized in the everyday world and are connected to the positions of the earth and the sun in space
3. The alternation of darkness and lightness on the earth is seen to have resulted from the turning or revolving of the earth in the earth's orbit.
4. The alternation of darkness and lightness and the succession of seasons has been seen to have resulted from the rotation of the earth around its axis once every 24 hours while the earth is revolving around the sun, a circuit around the sun taking place once a year.
5. In conceptual models which before the use of the simulation, the sun was revolving around the earth in two cases out of three the sun did not revolve anymore around the earth after the use of the simulation.

The children's exploration process is examined next from the basis of their conceptual models before the use of the simulation. Children were divided into four groups according to what kind of conceptual models they had before the use of the simulation. Children's conceptual models before the use of the simulation formed a starting point from which the exploration of the phenomenon was activated. Children's conceptual models and changes with in them can be read more precisely in Kangassalo's article 1993b.

The first group consisted of children (3/11) whose conceptual models before the use of the simulation were quite vague and undeveloped. After using the simulation a significant change in their models was that the earth began to revolve around the sun. The exploration of the phenomenon by two of these three children took place quite randomly at the earth level. They looked into the space level now and again but returned almost by the same way back to the earth level. One of these three children explored the phenomenon at the earth level more extensively than the other two and looked into the space level quite often. In this child's conceptual model after the use of the simulation the order, continuity and regularity of seasons in the everyday world had been discovered. The child used the simulation many times and her operating time was altogether one of the longest in the whole group of eleven children.

A little more developed were the conceptual models of four children (4/11). The order, continuity and regularity of seasons on the earth were quite well organized except one. Some light interrelations of phenomena on the earth and in space had been noticed. In these children's models, after the exploration process, the earth is revolving around the sun and the succession of seasons on the earth was perceived to be connected with the positions of the earth and the sun in space. The association of the alternation of lightness and darkness and the succession of seasons in the positions of the earth and the sun in space model proved to be problematic.

In this second group one child's conceptual model before the use of the simulation was somewhat uncertain about the succession of seasons on the earth but in his model the earth goes round. This child's exploration of the phenomenon was directed to phenomena on the earth. He explored very carefully and many times months at different points of the compass and systematically looked for details month by month. He looked into the space level very often almost in every season and used the telescope frequently to see the earth enlarged. The exploration of the phenomenon at the space level by the use of time characters took place only some times. After the exploration process in his conceptual model the succession of seasons were well organized both at the earth level and at the space level. The association of the alternation of lightness and darkness and the succession of seasons at the space level did not take place.

Two children of this second group explored the phenomenon mainly at the earth level looking briefly into the space level. In their models the earth was revolving around the sun and seasons on the earth were perceived to be connected with the positions of the earth and the sun in space. Connections in their models were still quite weak.

The fourth child of this second group explored the phenomenon the longest time (112 minutes) out of the whole group of eleven children. He started to explore the phenomenon on the earth. At first he l ʏ ːed for details according to seasons and months. Gradually he used the icons of the clock and days and after that he moved to explore at the space level. At the space level he explored the movement of the earth by using the clock, days and months. Sometimes now and then he looked into the earth. The child's exploration was systematic and took place with aim. Some degree of experimenting also occured especially at the latter part of the exploration process. After the use of the simulation in his model the earth is revolving around the sun and the succession of seasons on the earth was perceived to be connected with the positions of the earth and the sun in space. The association of day and night and succession of seasons at the space level did not take place. In his model these things were separate, so that daylight on the earth, in Tampere, is when the earth is on the other side of the sun and dark when it is on the opposite side of the sun.

The third group consisted of one child whose conceptual model of the phenomenon was very well developed before the use of the simulation. The only undeveloped thing in his conceptual model was the reconciliation of the alternation of lightness and darkness and the succession of seasons with the positions of the earth and the sun at the space level. During the exploration process the reconciliation was made. This child explored the phenomenon quite thoroughly both on the earth and in space. When using time icons, season, month, hour and day, the exploration took place side by side on the earth and in space.

The fourth group consisted of three children (3/11) whose conceptual models contained a clear misconception. In their models the sun was revolving around the earth. After the use of the simulation, two of the children's conceptual models became disintegrated. The sun's revolution around the earth was absent and interrelations of phenomena on the earth and space became weaker. One child's model remained almost unchanged. From these children's exploration there can not be seen any uniformity. One child whose model after the exploration process was disintegrated explored the phenomenon noticeably at both the earth level and the space level by using the time icons. The other child's exploration was quite random wandering and amusing. The child whose model remained almost unchanged used mainly the season and month icons on the earth level. Now and then he amused with the space shuttle and looked into the level of the solar system.

Summary about results

Children's conceptual models before the use of the simulation formed a starting point from which the exploration of the phenomenon by children was activated. Children's exploration processes seemed to contain wanderings here and there, investigating and seeking for something and experimenting with aim. Sometimes children amused themselves with the space shuttle and some children made tales about animals. The more developed and integrated conceptual model, the more children's exploration contained investigating and experimenting with aim. Children's conceptual model of the phenomenon develops and constructs during the exploration process in varying amounts but the main direction of constructing tended to be the same and seemed to follow the currently accepted scientific knowledge.

References

Kangassalo, Marjatta (1991). PICCO. The Pictorial Computer Simulation of a Selected Natural Phenomenon for Children's Use. Computer Program. Programming is done by Luv Kaul. Pictures are realized by Aila Penttilä.

Kangassalo, Marjatta (1992). The Pictorial Computer-Based Simulation in Natural Sciences for Children's Use. In Setsuo Ohsuga, Hannu Kangassalo, Hannu Jaakkola, Koichi Hori and Naoki Yonezaki (Eds.) *Information Modelling and Knowledge Bases III: Foundations, Theory and Applications*. Amsterdam: IOS Press.

Kangassalo, Marjatta (1993a). A Child's Conceptual Model of a Natural Phenomenon: Its Elicitation and Description. Forthcoming.

Kangassalo,Marjatta (1993b). Changes in Children's Conceptual Models of a Natural Phenomenon Using a Pictorial Computer Simulation as a Tool. In the *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*, June 18-21, 1993, Boulder, Colorado. Hillsdale, NJ: Lawrence Erlbaum.

Acknowledgements

I express my great thanks the Academy of Finland and the University of Tampere for the financing of my research. and the Department of Education, University of Tampere, for the excellent working circumstances.

Using Broadcast Journalism to Motivate Hypermedia Exploration

ALEX KASS, SCOTT DOOLEY, FRANK LUKSA, CHRISTINE CONROY
The Institute for the Learning Sciences
Northwestern University, Evanston, IL 60201
E-Mail: kass@ils.nwu.edu

Abstract: We describe a computer program, called **Broadcast News**, which teaches social studies to high school students by allowing them to put together a television news show. We attempt to show how the journalistic task, as structured by our program, can be used to motivate students to use an embedded hypermedia system to ask questions about social studies issues, to situate the knowledge that the students acquire, to challenge students to put this knowledge to use.

Introduction

In recent years, many developers of multimedia systems have focused primarily on the ways in which such systems can enhance a student's access to information. This is understandable, since easy access to information is a crucial element of effective learning environments. However, if the objective is for students to develop a working knowledge of a subject, then providing access is not enough. An effective learning environment must also ensure that students are motivated to access whatever information is available, that they have an authentic context (Collins, Brown, & Newman, 1989) in which to situate that knowledge, and that they face challenges that allow them to put that knowledge to use. Knowledge that is never put to use during the learning process may be knowledge that is difficult to retrieve and put to use when the need arises in the real world (diSessa, 1988).

In an attempt to address these issues, we have developed a computer program, called **Broadcast News**, which teaches social studies to high school students by allowing them to put together a television news show. We believe that the journalistic task, as structured by our program, does a very effective job of motivating students to use an embedded hypermedia system to ask questions about social studies issues, of situating the knowledge that the students acquire, and of challenging students to put this knowledge to use.

Overview of the Broadcast News System

The learning environment provided by Broadcast News has two distinct layers: a task-environment layer, and an instructional layer.

The task-environment layer consists of a miniature on-line production facility that students use to create a news show. In order to provide students with the ability to create TV-style news, Broadcast News requires some special hardware. The hardware configuration includes an IBM PS/2 with a video card for on-screen video, a laser disc player for video source material, and a video camera and computer-controlled VCR to capture the shows that the students create on VHS tape with the students as anchors. The source material in this layer includes footage from the CNN video feed and text from the AP wire-service. These sources are specific to the news days that the students will be working on and are necessary to put together a show.

The task-environment layer also includes the initial rough drafts of the stories on which the students will be working. These rough drafts are created by the curriculum designer and are constructed to focus attention on the crucial issues in the story. Rather than allow students to write their own, complete stories, students are only allowed to edit these drafts. By giving students a finite number of changes that they can make, the program is able to carefully follow what they do. Providing rough drafts also allows the curriculum designer to structure the learning experience. To provide the appropriate challenge, curriculum designers intentionally place errors and biases into the rough drafts, leave important things out, choose confusing terms, etc. These challenges are intended to raise questions in students' minds. By anticipating these questions and by making the knowledge base easy to explore, the editing task can encourage students to seek out the information required to answer these questions. (For a discussion of our techniques for structuring the editing task, see Kass, et al., 1993).

The second, instructional layer of Broadcast News provides expert feedback and additional sources that students can access to get the information they need to make editorial decisions. When students lack the background knowledge necessary to understand the issues raised in a story, help is available in hypermedia form. The system contains video of subject-matter experts answering relevant questions. In addition, background material from standard reference works (text and video) is available. A portion of the screen is devoted to displaying questions which students can ask that are relevant to the currently selected text of the story. By clicking on a question, students can bring up an expert who will answer the question, or even multiple experts with different points of view on the question. At any point while the answers are being played, the students can stop the video, and either go back to the editing task, or ask a follow-up question. For students who may fail to notice problems on their own, the program provides some prodding in the form of expert objections to some of their editorial decisions or to the approved draft. The experts also offer unsolicited comments challenging the students' editorial decisions.

Broadcast News allows a curriculum designer the freedom to choose stories that touch on important issues, to structure the students' interaction with the program, and to provide appropriate support material, challenging students and providing the information they need to accomplish their task.

Using Broadcast News

It takes a team of people to put together a real television news show, including producers, writers, video editors, and anchors. During each session with Broadcast News, the student plays some of these roles, and the rest of the roles are played by the computer. In the current version of the program the student's main task, which corresponds roughly to the job of an assistant producer, is to edit the text and video of the rough draft.

When editing a rough draft (Figure 1) the screen is composed of the following items:

- A rough draft of the story, including markers that represent when video will be shown.
- A list of the on-line source material (text files and video segments) related to the story.
- A list of questions about the story that are answered by either text or video clips of experts.
- A collection of relevant hypermedia reference works.
- A set of control buttons the student uses to take editorial actions, such as marking up the story for rewrite, asking for the rewrites to be executed, and giving final approval to the story.

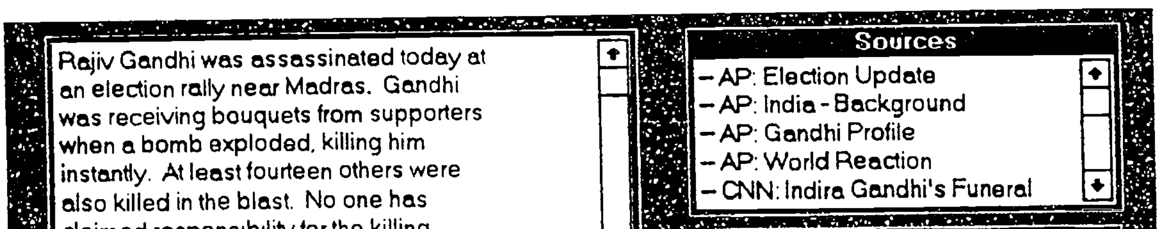


Figure 1. Broadcast News Editing Screen

After the main editing task is complete, and the show is ready to "go on the air," the student is allowed to play anchor. The editor's job is the one that challenges the student to think about the issues behind the story; most of the learning is expected to take place during that phase. The anchoring phase is included because students generally enjoy it, and because it allows them to produce a videotape of their show, which can motivate them to invest more of themselves in their work.

An Example of Broadcast News in Action

A typical session with Broadcast News, in which the student plays the role of assistant producer, begins with a note from the boss, the executive producer, describing the day's assignment. The student's job is to edit the lead story for May 21, 1991, the day that Rajiv Gandhi, the former Prime Minister of India, was assassinated.

After reading the memo from the boss, the student proceeds to the main screen (Figure 1). Students generally begin by reading the rough draft of the story. The first sentence of the Gandhi story reads: "Rajiv Gandhi was assassinated today at an election rally near Madras." Many secondary-school students do not fully understand the references in this sentence. Many do not know who Rajiv Gandhi was or where Madras is. To alleviate this problem, the student can turn to the story questions.

Asking Questions of the Experts

Our prototype system includes video clips of two experts, both Northwestern University professors. One is a former president of NBC and professor of journalism. The other is on the political science faculty, and is the author of a very popular college-level political science textbook.

When no particular portion of the story is selected, the system offers questions that are general to the entire story. The box in the lower-right corner of Figure 1 depicts this situation. One question available is, "What makes this story newsworthy?" Clicking on this question brings up a video clip of our journalism expert, explaining that Gandhi was a world leader and that news is the reporting of change, such as the death of a world leader. While the answer is being delivered, the question list changes to present follow-up questions.

In addition to general story questions, specific questions pertaining to portions of the story are available. By selecting a range of text in the story, the student tells the system to bring up a set of questions related to items mentioned in the selected text. In the Gandhi story, if the student selects the first sentence, the system will bring up a set of questions about Gandhi, Madras, etc. One of the questions is "Who was Rajiv Gandhi?" If the student asks this question, the political science expert explains that Gandhi was a former prime minister of India, who was running for re-election. Follow-up questions a student might want to ask are: "What does it mean to be a prime minister?", "What is an assassination?", What was Gandhi's political platform?", and "Why was Gandhi running for office again?"

The student may also click on the video marker to see questions that relate to the video used in the story.

Reviewing Source Material

Before editing a story, it is usually a good idea to look at the source material for the story. Selecting a source from the list makes it display on the screen. One of the wire service articles for the example story is titled "Gandhi Profile". The example text article tells the location of Madras in India and more about the Gandhi family and how long they have ruled India.

Video source is available as well as text. In addition to providing background knowledge, the video source can be inserted into the broadcast in place of the video that is currently designated to be included in the story.

Editing the Story

After asking any introductory questions, the student should be ready to make some edits. Choosing a sentence and clicking the Edit button brings up a menu of the following rewrite options: Add to selected text; Change selected text; Delete from selected text; or Revise entire paragraph (Figure 2).

In the example story the opening sentence has very little detail. The student may want to add some information about Rajiv Gandhi. Once the student has chosen what type of edit to make, a new menu appears detailing the edits that are available. For example, choosing "Add to selected text" brings up a menu of the following more specific choices on which the student can add detail: Gandhi's family; Gandhi's Title; or Location of Assassination (Figure 2).

It is important to note that none of the choices is intended to be *right* or *wrong* in any absolute sense. Different students will make different choices, just as different television stations make different choices. The point is not to test the student to see if he or she can come up with the right answer; the point is to require thinking about the issues and making decisions. However, the fact that there is no definitive right answer does not mean that the experts do not have their preferences. If the student opts for adding detail about the method of assassination, the political scientist will object: "I think that the details of the assassination are less important than what this means for the future of Indian democracy." But the fact that an expert disagrees doesn't mean the student has made the wrong choice. The student may find that the experts have incompatible opinions. The students might get challenged no matter what choices they make. The point of the challenges is to make sure students think about what the important issues are and reach an informed decision.

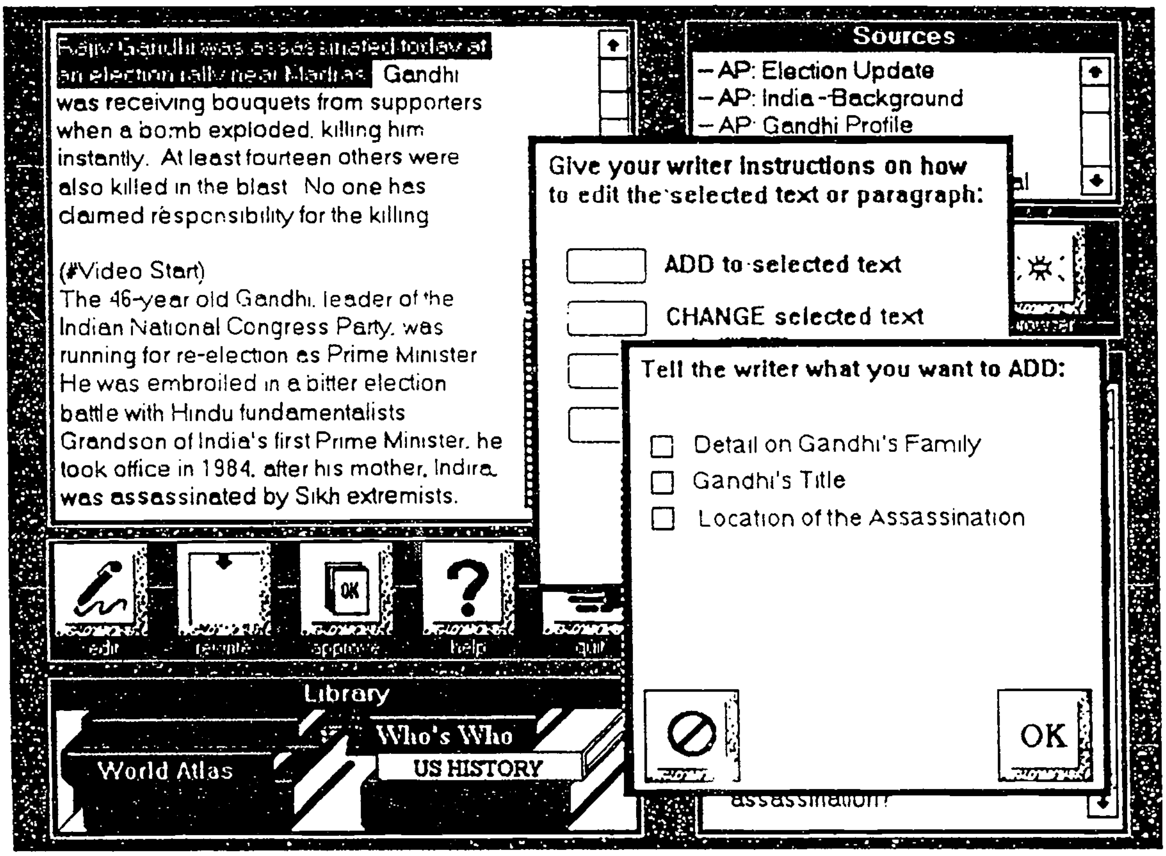


Figure 2. Rewrite Options

After a student chooses to make an edit, such as "More on Gandhi's title", the selected text gets "marked up" by having its color changed to red. The student can also make changes to the video clip used in a story. The edited text or video will remain marked until it has been sent back to the writer to have the prescribed changes made.

Rewriting the Story

Students may, at any time after making an editing decision, choose the Rewrite button and send the marked text and video to the programmed writer for a rewrite. The story returns with changes made as directed. In the example story, if the student has asked for more detail about Gandhi's title in the lead sentence, it will come back transformed from "Rajiv Gandhi was assassinated..." to "Former Indian Prime Minister, Rajiv Gandhi was assassinated...".

Submitting the Story for Expert Review

After receiving the rewritten text, students may make more editing decisions, do more research, or decide that the story is ready for broadcast. When the student decides the story is good enough to go on the air, she chooses the Approve button. Upon student approval the program checks with each expert for comments on the story.

Regardless of how well the rough drafts are crafted, students cannot always be expected to address all the points in the story that the curriculum designer feels are important. In these cases, after the story is approved, one or

 333

several experts will explicitly raise the issues by commenting on them. Although the student is not required to take action on any of the comments, it is expected that they will often serve as motivation to reread the story and consider the issue.

Anchoring the Story

After giving final approval for a story, the student next needs to anchor and videotape it. For this, the program provides a mini-studio (Figure 3) including a camera, programmed teleprompter, and video viewing window. Just as in a real studio, the student can practice the story first, starting and stopping the teleprompter and seeing how the video will work with the story. Unlike a real news show, the student gets to decide when she is ready to start anchoring. When ready, the student clicks the Go On The Air button. The introductory graphics announcing the news show roll and then the teleprompter starts. The student reads the story as it scrolls by in the teleprompter. Upon reaching the part of the story that indicates the video should start, the student clicks a button to start the video, then chooses another to switch the video source back to the camera. When finished taping, the student can view the show.

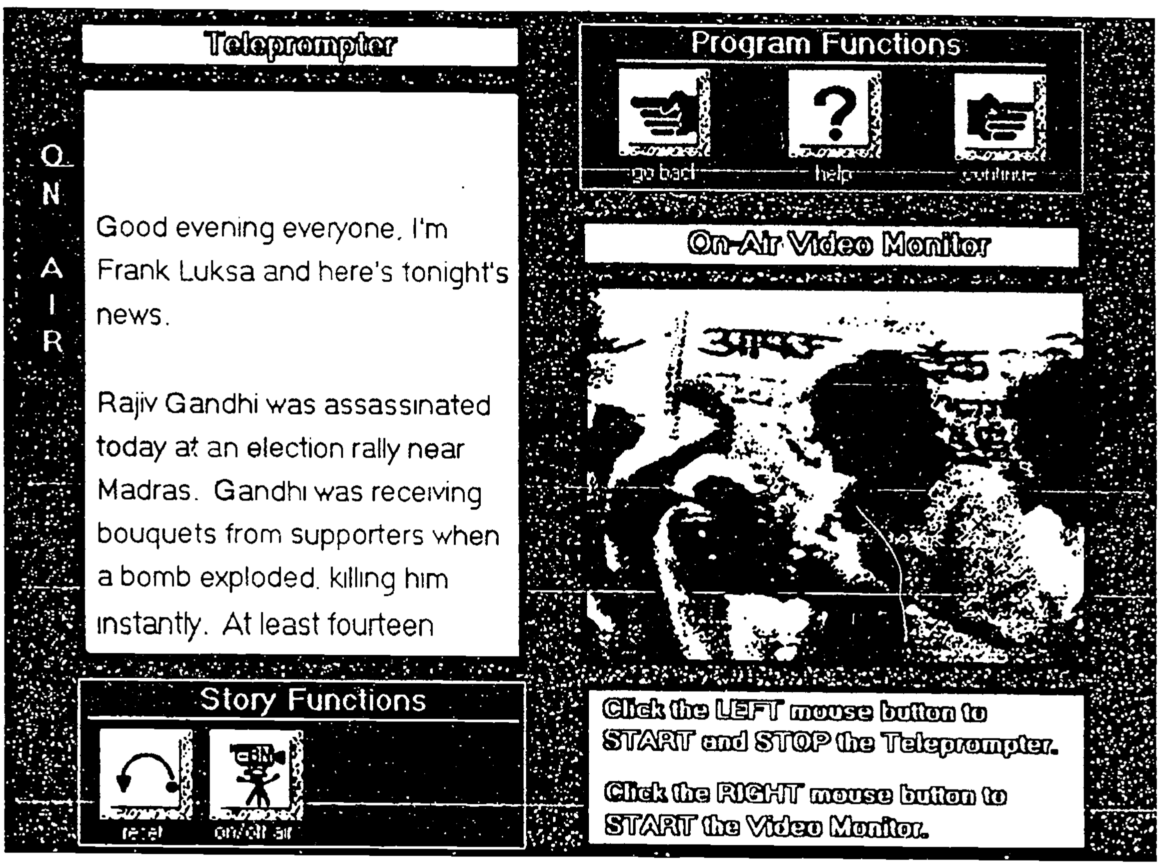


Figure 3. Broadcast News Studio

Viewing the Show and Finishing Up

In the viewing screen, the student is presented with a video monitor cued to the beginning of the introductory graphics that start her news show. In addition to watching her show, the student may also choose the NBC News button to see what the network decided to do that night. In our example news day, the student sees that NBC also chose to use the Gandhi assassination as the lead story. The comparison between what the network put together and what various students put together is a good starting point for a class discussion.

After completing all the steps for a news day, the student can take the videotape of the show with her. As students complete more news days, they get more responsibility for stories and the show as a whole. Students can look forward to eventually being able to write their own stories starting from only the source material.

Current Status and Future Plans

Work on the Broadcast News system is proceeding on three fronts in parallel: 1) extending the application; 2) developing a curriculum based on the use of the program; 3) extending the authoring tool that facilitates the creation of new Broadcast News sessions. In addition to our work on developing Broadcast News, we are also evaluating the system under controlled experimental conditions at a nearby high school.

The Broadcast News program is at the stage of an advanced prototype. Students can use the program in the assistant producer role, developing a small number of stories. However, the current version of the program lacks some features that we deem important. For instance it does not allow the student to review the events that took place following the date of the news day the student worked on. Getting a recap of events is important to allow a student to understand how events unfolded.

Also, the program currently provides a rather limited range of reference material: Dictionary entries defining words in the rough draft, and the AP and CNN news services. A broader range of materials is needed.

We are currently developing a course on the First Amendment to the U.S. Constitution. The course will consist of a series of Broadcast News sessions, all of which touch on a First Amendment issue. Through a set of stories addressing issues such as the burning of a U.S. flag, censorship of high school newspapers, and the use of allegedly obscene music lyrics, we believe we can bring alive the conflicting needs for both freedom and order in our society. To produce good stories in this unit, students will need to know fundamentals about our government such as the organization, responsibilities and interactions of government branches, as well as the content of the U.S. Constitution, how it was formed and how it can be amended.

In addition to work on the application, we have been developing an authoring system to allow a content expert to construct new Broadcast News sessions by entering rough drafts, background information, and expert commentary into the system. The content specialists on our development team are currently using an initial version of this tool. Eventually, educators should be able to use the tool to develop new Broadcast News sessions without needing sophisticated computer skills.

Conclusion

We believe that providing an authentic, exciting activity is the best way to stimulate learning. The activity serves to motivate students, and also to provide them with a context in which they can use the information made available to them. The task provided by Broadcast News - editing a TV news story - seems to be a good one for these purposes for the following reasons: students are excited by the task; the task allows them to produce a concrete artifact in which they can take pride; and the task can be structured to focus students' attention on specific social studies issues. In this context, the expert commentary we have captured on video disc and placed in a hypermedia knowledge base can be used to best effect. All instruction is presented at precisely the point in the student's activity when it is directly relevant to something the student wishes to do.

To summarize, here is our recipe for an effective learning environment: Provide students with an exciting activity that is carefully structured to focus their attention on the issues you wish them to learn about, and provide them with the expertise and the information resources needed to answer the questions that will naturally arise when they attempt to perform the activity.

References

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the Crafts of Reading, Writing, and Mathematics. In L. B. Resnick (Eds.), Knowing, Learning, and Instruction: Essays in honor of Robert Glaser (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum Associates.

DiSessa, A. (1988). Knowledge In Pieces. In G. Forman and P.B. Pufall (eds.), Constructivism In the Computer Age. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kass, A., Dooley, S., Luksa, F. (1993) The Broadcast News Project: Using Broadcast Journalism as a Vehicle for Teaching Social Studies. The Institute for the Learning Sciences, Northwestern University.

Acknowledgments

Many people have contributed to the design of Broadcast News. In addition to the authors, the major contributors include Noreen Burke, Michelle Saunders, Diane Schwartz and Josh Tsui. Robert Mulholland, our journalism expert, Jerry Goldman, our political science expert, Alan Dershowitz, Professor of Law at Harvard Law School, and Jay Miller, Executive Director of the Illinois branch of the American Civil Liberties Union, have given generously of their time and expertise. This research was funded in part by a grant from IBM. The Institute for The Learning Sciences was established in 1989 with the support of Andersen Consulting. The Institute receives additional support from Ameritech, and North West Water, Institute Partners.

A Knowledge-based Multimedia System to Support the Teaching and Learning of Chinese Characters

W. KI, S.K. TSE, N. LAW
Department of Curriculum Studies
University of Hong Kong, Pokfulam Rd., Hong Kong
Email: hraskww@hkucc.hku.hk

F. LAU, K.H. PUN
Department of Computer Science
University of Hong Kong, Pokfulam Rd., Hong Kong

Abstract: This paper reports on a current project to develop a multi-media system that would support the teaching and learning of Chinese characters, as well as provide a platform for conducting research into the cognitive aspects of Chinese language acquisition. Although the number of Chinese characters commonly in use amounts to thousands, there are in fact many structural commonalities and regularities among the characters, which relate these characters into a highly connected system. One of the objectives in developing the present system is to build a knowledge-base to represent these structural features of and relationships among Chinese characters, so as to support the development of interactive learning environments where these underlying structures of Chinese characters can be made more explicit and explorable to the learner. It is envisaged that through interactions with such environments, the learner would be able to develop a structural understanding about Chinese characters and acquire effective strategies for learning them.

This project intends to help young first language learners, from pre-primary to early primary levels, to learn Chinese characters. In learning a character, the learner has to learn its written form, its pronunciation and its meaning. There are reasons to believe that before the school begins to teach these children the written form of Chinese, the children have already acquired, from their daily life, a rich mental lexicon for the language, in its audio form (Kwong 1992). So the major tasks to be achieved in teaching these characters are (1) to provide effective means for the learner to encode the written form of the characters, and (2) to help the learner to relate these written forms to sounds and meanings already existing in their mental lexicon (Huang & Liu '978). In the following sections, the authors will examine how a computer system may provide support for this teaching and learning process.

Effective Learning of the Written Form of Chinese Characters

Ideographic Nature of Chinese Characters

Chinese characters are not just arbitrary aggregates of strokes. Many Chinese characters are pictographs standing for objects, or pictographs with certain markings added to indicate more abstract concepts (Shuowenjiezi, 100,Weiger 1911; Hung & Tseng 1981).

Research shows that at an early stage in the learning of Chinese characters, the learner tends to remember characters as distinct pictures (Chuang 1975). In one earlier project on software for learning Chinese characters carried out by the team (Lam 1993), computer animation was used to relate the written form of some Chinese characters to their pictorial origins. Preliminary evaluation indicated that such presentation is motivating as well as effective in helping young learners to remember the written form as well as the meanings of the characters.

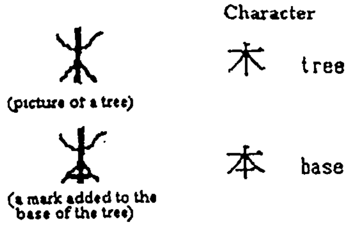


Figure 1. Chinese characters as pictorial and indicative symbols

Chinese Characters as Structured Compositions

Most Chinese characters are composite logographic forms (Shuwenjiezi, 100). In these "compound characters", two or more major components are combined to fill an imaginary square block. The methods of composition are highly regular. The components are reusable in producing different characters, and many of them are themselves characters. Some basic methods of composition are shown in figure 2, each illustrated with a compound character of that format.

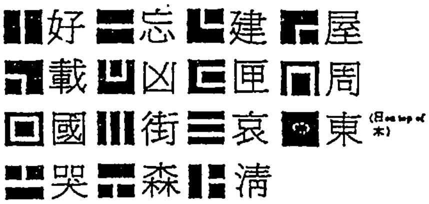


Figure 2. Chinese characters as structured compositions

In general, the components contribute to the sound or the meaning of the compound character they form (Ann 1982). In some cases, the components are themselves characters, and the meanings of these simple characters add together to form the meaning of the compound character.

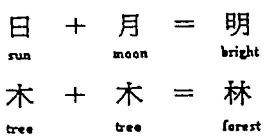


Figure 3. Characters representing an aggregate of meanings contributed by its components

In other cases, one component represents the meaning while the other component the sound of the compound character. This is the so called pictophonetic composition. Over 70% of Chinese characters are compositions of this sort.

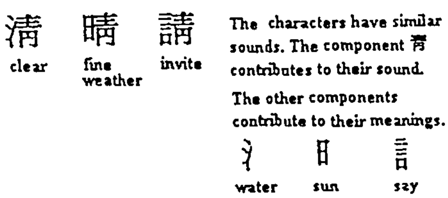


Figure 4. Examples of characters formed by pictophonetic composition

If one further examines these components, one will notice that they can in fact be further decomposed into simpler reusable stroke patterns. These we shall refer to as "graphemes". Although this decomposition to the graphemic level often does not add knowledge about the meaning or sound of the character, familiarity with the set of commonly used graphemes do help people in their recognition and writing of the characters or components (Huang 1992).

Structured Encoding of Chinese Characters by Learners

In studies conducted by Shen Yeh and Tsao Chuan Yung (1963, 1965), it was found that there were two major developmental changes in the process of mastering Chinese characters by primary school children: the first one was found immediately after the first month of learning Chinese characters and the other was between the first and second grades. With the second transition, the ability to master unfamiliar characters grew rapidly.

According to reported experience of some local school teachers, learners at grade two begin to demonstrate some mental processes of graphemic analysis and synthesis. Having learnt the characters 兄 (brother) and 兒 (son), a learner may identify the reusable grapheme "儿". When presented with a new character 插 (insert), the learner may be able to describe verbally the component 臿 as a 千 (thousand) on top of "a 兒 without the '儿' ". This ability to generate new logographic constructs to chunk new information would help to reduce significantly the memory load required in encoding the form of new characters (Hung & Wang 1992). Furthermore, some learners can guess that the character means some action, because they know other characters like 打 (hit), 推 (push), 拉 (pull) also have the same component 扌 and they all mean some actions.

Using Computers to Facilitate Structural Perception and Understanding of Chinese Characters

Although research seems to show structural encoding to be an important process in the learning of Chinese characters, current practice in primary grades is to drill learners in the stroke-by-stroke writing of characters (CDC 1990). Much of the learner's attention is thus directed to the individual strokes, while the development of structural perception of Chinese characters is very much left to his/her own initiative.

In this project, we see great potentials for the computer to provide useful learning experience, beyond the conventional handwriting exercises, for the learner to develop a structural perception of Chinese characters. The computer, with its interactive graphic capability, can provide a medium where character structures can be presented with flexible highlights directing learners' attention to the various components. The computer can also provide a rich variety of activities where the learner can practise disassembling and assembling logographic components of Chinese characters.

More importantly, the computer can allow the learner to learn the written form of Chinese characters with more meanings and interconnections. As pointed out earlier, there are many regularities in the construction of Chinese characters and the rich information encoded in the written form of a Chinese character may include cues about its meaning and sound. Hypertext systems can be designed such that characters related through particular structural features or components can be made readily accessible for the learner to examine. Characters can be learned in rich relationship with one another and learners can traverse from one character to the next through a shared component. Learners may even explore, like linguists, the regularities in the structure of Chinese characters. Knowing about the components of a character becomes an exciting pathway for information access and concept formation. In exploring this hypertext, learners can construct their own theories about the structure of Chinese characters. Such an approach to the learning of Chinese characters will be referred to as the *structural constructivist approach*.

Although Chinese characters are ideographs and meanings are attached to individual characters, such meanings are often not too specific. Most nouns, verbs and adjectives in Chinese are in fact made up of two or more characters. Such linguistic units will be referred to here by their original name in Chinese: *ci* (詞). In many instances, *ci* seem to correspond more to words in English, while Chinese characters to monosyllabic morphemes (Hoosain 1992, Huang & Liu 1978). *Ci* ususally have meanings much more specific than characters. For example, the idea of a policeman, corresponding to the *ci* 警察, is concrete and commonplace in everyday life, while the character 察 carries a more general or abstract notion of to notice or to examine. A possible way for young learners to learn the meaning of a character would be to explore the meaning of a number of commonly used *ci* containing the character and make their own induction. Here the hypertext capability of computer can again play an important role.

A Knowledge-based Multimedia System to Support the Learning of Chinese Characters

This section describes the design of the system we are currently developing. It is intended to be a flexible and extensible platform for the development of computer-aided-learning (CAL) software for Chinese character learning, as well as research into cognitive aspects of Chinese language learning. The system is designed to support, in particular, teaching and learning in the structural constructivist approach.

Conceptually, the system consists of five levels. The bottom layer of the system is a knowledge-base about structures, relationships, and meanings of Chinese characters. The second level is a database which records the frequency, extent of use and competence of a student in relation to the different elements in the knowledge-base. The third level is a set of application software providing access to the Chinese character knowledge-base (first level) and the student progress database (second level) for use by teachers and students. The fourth level is a set of courseware generation tools for use by teachers to generate customized CAL software. The fifth level is a set of CAL materials developed by tools in the lower levels for teaching various topics about Chinese characters.

The System Backbone: The Chinese Character Knowledge-base and Its Basic Functions

The Chinese character knowledge-base is built to represent the structural features and relationships among Chinese characters. It consists of two modules, one responsible for the logographic information of characters, and the other responsible for the meanings and sounds of characters.

The logographic information module contains a database of logograms, including characters, components, graphemes and strokes. It records how each logogram is stepwise-decomposable into other simpler logograms, and the composition method. In the case of pictophonetic composition, it also makes note of the specific subcomponents contributing respectively to the sound and the meaning of the character. Pictorial origins of characters are stored as graphic animation where appropriate.

The information stored will support the following operations:

(a) to give a stroke-by-stroke calligraphic display of the character.
(b) to provide a stroke ordering exercise where the learner can choose the correct stroke sequence in a character with continuous feedback from the computer.
(c) to display the character, highlighting in turn the different components and subcomponents.
(d) to support jigsaw-like exercises where the learner is asked to assemble characters from their components.
(e) partial matching to search for a character based on partial information about its structures and components

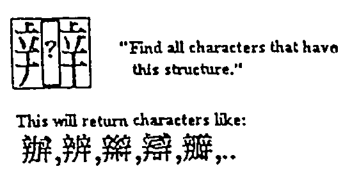


Figure 5. Example of a search based on partial information on the written form

(f) to present the pictorial origins of some characters. The presentation starts with a realistic picture, and gradually transforms to a skeleton, and then to the written form of the character. The method is similar to that used in Lam (1993)

In *the sound and meaning module*, each character node is linked to a set of *ci* illustrating the meaning of the character. A voice file and a phoneme are stored for the pronunciation of each character in the knowledge-base. For the special cases where a character is pronounced differently when it is used in different senses, a number of voice files and phonemes will be stored. For some of the *ci*, additional resources are stored including images and voice files of sample sentences. Further, for each character and *ci*, there is information about its meaning category and difficulty level.

Such information is to support the following functions in this module:

(a) to display related *ci* to illustrate the meaning(s) of the character.
(b) to read aloud the characters and *ci* in the knowledge-base.
(c) to read aloud example sentences or display pictures for a particular *ci*.
(d) to search for characters that are pronounced identically.
(e) to search for characters based on partial information on phonemes.

(f) to search for *ci* according to their meaning categories, and difficulty levels.

To start with, the system will contain information for 500 Chinese characters selected from those specified in the school curriculum for young first language learners (CDC 84, 90). According to our analysis, we envisage that within these 500 characters, the structure and basic graphemes necessary for the learning of most Chinese characters would have been covered. We hope that by learning these 500 characters in the way supported by the system, the mental "tool-kit and library" can be developed so that the learner will be able to learn other Chinese characters proficiently (Shiu & Lau 1982, Tseng & Wang 1983).

The Other Levels of The System

The second level - the student progress database. The limitation of any teaching/learning material that does not take the learner's past experience and competence into account is widely recognized. The objective here is to set up a database that records for each student his/her usage profile and achieved competence as demonstrated by his/her interactions with the system. Data will be automatically logged into this database when the learner uses the various modules at the higher levels. This database can be used to provide feedback to students, to allow teachers to monitor the progress of students, and for use by the software generators at level 4 to generate appropriate CAL software.

The third level - database tools. This is a set of application software that allows both teachers and students to modify, customize and access the Chinese character knowledge-base and the student progress database.

For the teacher, there are four types of application software:

1. The knowledge-base construction tool - This is a user-friendly interface provided to allow modification and extension of the knowledge-base. Teachers may add in new characters, structures, sense nodes, *ci*, together with voice and image resources, and even new meaning categories. This is seen as an extremely important feature. It means that teaching of Chinese characters through the system need not be confined by the exact contents of the knowledge-base as provided. Teachers can change it and such changes can automatically be reflected in all the applications built on the knowledge-base.
2. The teacher query tool - This is a tool provided to teachers to query the knowledge-base for lesson planning and courseware development purposes.
3. The teacher customization tool - As with printed dictionaries and thesauruses, different versions will be required for different levels of users. The customization tool allows the teacher to index the knowledge-base into different levels of difficulty and allow student access to a designated portion only.
4. The reporting tool - This is for teachers to monitor the progress of individual students. It reports on the profile of usage: modes of usage, parts of the knowledge-base traversed, and performance in cases where the student has taken exercise or tests (level 4 and 5 applications) from the system.

For the student, there are three types of tools:

1. The student query tool - It may be used as a general reference tool by the student. The basic functions have been outlined in the previous section about the knowledge-base. Teachers may design specific learning tasks or research projects that require the use of the knowledge-base, or students may be allowed to call up this tool for help while they are working on other software on the system. .
2. The student customization tool - The purpose of this tool is to allow the student to create a personalized knowledge base of Chinese characters and *ci* that he/she wants to retain for easy access in future. Furthermore, the student may construct his/her own data-structure for organizing language elements (which in itself offers great potential for research into cognitive features in Chinese language learning).
3. The reporting tool - This is similar to the teacher reporting tool. It allows the student to find out how much he/she has learnt and mastered for self-monitoring purposes.

The fourth level - CAL generation tools. This is a set of software generation tools that can produce courseware in specific formats. The teacher just has to choose the courseware type, and specify the required language elements to be used, how he/she would like to draw reference from the student progress database, and how the student progress database should be updated according to the student's performance. The courseware will be automatically generated. One would envisage instruction like the following would be sufficient to generate a CAL program: "Generate a program on assembling characters with two components, in the form of a cat-catch--mouse game, using the set of characters in the knowledge-base that has been visited by the learner. Add one point to the learner's structured-perception index to a character if he/she succeeds in assembling the character."

The fifth level - CAL materials. The fifth level is a set of example courseware and self-learning materials, both computer-based and non-computer based, developed using the tools provided at the lower levels. It illustrates how the generic tools in level 4 can be use to achieve specific learning objectives. It includes learning materials that introduce new characters and *ci*, materials that focus on particular topics about Chinese characters or particular language skills, games for enhancing interest or for reinforcement, and tests. While these courseware are being used on the computer, the system will be able to update the student progress database at the second level of the system. Also, the learner can have the option of moving from the courseware to the student query tool for reference or help.

Conclusion

This paper reports on the considerations made in the development of a multi-media system that would support the teaching and learning of Chinese characters for young first language learners. The logographic and morphemic nature of Chinese characters are discussed, and a structural constructive model of learning is described. The knowledge-base is currently under construction and the application programs are now in the process of detail design. It is envisaged that the evaluation of such a system in classrooms is going to provide useful feedback on the various theoretical assumptions about the cognitive processes in Chinese language learning discussed in the paper.

References

Ann T.K. (1982), *Cracking the Chinese Puzzles*, H.K.:Stockflows Co., Ltd.

Atkinson,R.C. (1975), Mnemonics techniques in second-language learning. In *American Psychologist*, Vol. 30, pp. 821-828.

Chuang C.J. (1975), The function of imagery in learning of Chinese language, *Acta Psychologica Taiwanica*, Vol. pp.145-150.

Hong Kong Curriculum Development Committee (CDC) (1984), *Guide to the Kindergarten curriculum*, Hong Kong Education Dept.

-- (1990), *Chinese language syllabus for primary schools*, Hong Kong Education Dept.

Hoosain, R. (1992), Psychological reality of the word in Chinese. In*Language processing in Chinese*, H.C. Chen & O.J.L. Tzeng (eds). Amsterdam: Elsevier Science Publishers

Huang, J.T. & Liu I.M.(1978), Paired-associate learning proficiency as a function of frequency count, meaningfulness, and imagery value in Chinese two-character ideograms. *Acta Psychologica Taiwanica*,1 67, 5-17.

Huang J.T. & Wang M.Y.(1992) From unit to gestalt: perceptual dynamics in recognizing Chinese characters. In *Language processing in Chinese*, H.C. Chen & O.J.L. Tzeng (eds). Amsterdam: Elsevier Science Publishers

Hung, D.L., & Tseng, O.J.L. (1981) Orthographic variations and visual information processing. *Psychological Bulletin*, 90, 377-414.

Kwong, S.M. (1990) *The syntactic development of Cantonese-speaking preschool children*. M.Ed. Thesis, University of Hong Kong.

Lam.H.C., Pun,K.H., Leung,S.T., Tse,S.K. & Ki,W.W. (1993), Computer Assisted Learning for Learning Chinese Characters. In *Communications of COLIPS, an international journal of the Chinese and Oriental Languages Processing Society*, Vol. 3, No.1.

Liu Y.T. (1965), Experimental studies on the effects of pictorial material and writing in learning characters, *Acta Psychological Sinicia*, Vol. 3.

Paivio,A. & Casapo,K. (1973) Picture superiority in free recall: imagery or dual coding? In*Cognitive Psychology*, Vol. 5, pp 176-206.

Shiu P.K. & Lau S. (1982), The association pattern of Chinese characters of Hong Kong kindergarten children. In *Psychological Studies of Chinese Language*, S.R.Kao & C.M.Chen (eds.) pp.103-122, Taiwen:Wenhe.

Shuowenjiezi (A.D.100), *The analysis of Chinese characters*. H.K.:Chung Wah (reprint)

Tsao C.Y. & Shen Y. (1965), Developmental studies on the recognition and generalization of Chinese characters in primary school children. *Acta Psychologica Sincia*, Vol 1 & 2.

Tzeng, O.J.L. & Wang W.S.Y. (1983), The first two R's. *American Scientist*, 71, pp. 238-243.

Weiger, L (1915), *Chinese characters, their origin, etymology and history*. Catholic Mission Pub.

Acknowledgment

The authors would like to acknowledge the contribution of the other team members in their contribution to the software development project: M.K.Lau, H.C.Lam, S.S.Lam, Y.T.Lee, W.H.Chau & W.K.Tam The project is sponsored by the Hong Kong University and Polytechnic Grant Committee research grant, 1991. (hku 33/91)

Digital Captioning: Effects of Color-Coding and Placement in Synchronized Text-Audio Presentations

CYNTHIA M. KING
Educational Foundations and Research
Gallaudet University, 800 Florida Avenue, NE, Washington, DC 20002-3695
email: cmking@gallua.gallaudet.edu

CAROL J. LASASSO
Education
Gallaudet University, 800 Florida Avenue, NE, Washington, DC 20002-3695
email: cjlasasso@gallua.gallaudet.edu

DOUGLAS D. SHORT
Institute for Academic Technology
2525 Meridian Parkway, Suite 400, Durham, NC 27713

Abstract: Captioning is the process of providing a synchronized written script (captions) to accompany auditory information. This article describes programs available for captioning digital media on computers and reports the results of a study on color-coding and placement of captions. Results indicate that comprehension is higher when captions are color-coded for speaker identification than when captions are black-and-white. There are no significant differences in comprehension between centered captions and captions with variable placement dependent on location of the speaker.

Captioning is the process of providing a synchronized written script (captions) to accompany auditory information. The most common form of captioning is that for *analog* television, where captions are embedded in Line 21 of the video signal and displayed only to those who have an external decoder or a television with a built-in decoder (Armon, Glisson, & Goldberg, 1992; Bess, 1993). New methods of captioning, however, are beginning to be developed for *digital* television and computer-based multimedia products (Armon et al, 1992; Hutchins, 1993; Short, 1992; Short & King, 1994).

King (1993) provides an extensive discussion of captioning and why multimedia developers should incorporate it into their products. Additional discussion of issues related to ensuring that deaf and hard-of-hearing people have access to information presented auditorially can be found in Jordan (1992) and Kaplan and De Witt (1994), as well as in electronic resources on the Internet, such as ADA-LAW (ada-law @vm1.nodak.edu) and Equal Access to Software and Information (easi@sjuvm.stjohns.edu).

The present paper addresses: (a) digital captioning, (b) captioning format research and standards, and (c) effects of color-coding and placement of captions used to represent speaker identification on the comprehension of deaf and hard-of-hearing viewers.

Digital Captioning

Synchronization of text and audio is possible in a wide array of computer-based environments. For example, IBM provides an option within the *OS/2* operating system by which captions, if they exist, can be turned on or off at the request of the end-user. IBM also provides a simple captioning tool, whereby developers can synchronize text files to accompany audio, as part of its Ultimedia Tools series. On both PC and Macintosh platforms, some video-editing programs support captioning via titling functions (e.g., ATI's *MediaMerge* and Adobe's *Premiere*; the titling function in *Premiere* is currently available only in the Macintosh version).

Multimedia programming environments, such as Apple's *Hypercard* and Asymetrix's *Toolbook*, can also be used for captioning. The research described in this paper involved the use of captioning tools developed, using Asymetrix's *Toolbook*, by Douglas Short of the Institute for Academic Technology. These tools were originally developed for providing translations for foreign language operas on laserdiscs (*SyncText*) and for providing notation and/or translation for music and lyrics on CD-Audio discs (*CD Time Liner*) (Institute for Academic Technology, 1992; Short, 1992).

The *SyncText* laserdisc captioning program was subsequently enhanced by Short and King (1994) to include tools for statistics calculation (e.g., words-per-minute) and other functions (e.g., a function to find words and phrases in the caption list) that are helpful during the captioning process. The enhanced tool, *CAP-Media LD*, also meets or exceeds the current specifications for television captioning. For example, the FCC Caption Decoder Standard of 1991 (as revised in 1992; Electronic Industries Association [EIA], 1992) mandates that seven background and foreground colors be available; *CAP-Media LD* provides twenty different colors. *CAP-Media LD* can align captions via all methods available for television (left alignment, eight tab-settings, and/or starting at any character in the 32-character line possible on television). It also provides automatic centering and right-alignment. Line length is not restricted in *CAP-Media LD*, and a full range of proportional and monospace fonts is available. The fonts include all characters in the required and optional font sets, as specified for television captioning (EIA, 1992). The maximum number of caption lines is set at four, identical to the limit in television captioning (in *CAP-Media LD*, this limit can be overridden manually, if desired).

CAP-Media LD, which works in conjunction with a set of tools called *Express Author* developed at the Institute for Academic Technology, permits the video and caption areas to be displayed full screen or sized to any portion of the computer screen. The program includes options for users to customize the target applications they develop using *CAP-Media LD*. For example, users can create lists of captions and graphic charts of video segments, which can then be used to control display of the video in non-linear order. Interactive captioning is also possible via hyperlinks between the captions and *Express Author*'s glossary tools. Thus, if a glossary and hyperlinks to the captions have been created, end-users are able to click on words or phrases as captions are displayed. The video is stopped automatically and the user is linked immediately to supporting materials--which can include text, video, audio, animation, and/or pictures--for clarification or elaboration. The glossary tools also permit end-users to add their own entries to the glossary.

Finally, *CAP-Media LD* is designed such that display of non-text objects (e.g., graphic symbols) and other *Toolbook* functions can be synchronized to audio-visual displays in the same way that text is handled. This design will allow, for example, synchronization between display of a laserdisc video clip and an audio file being used for descriptive video services (WGBH Educational Foundation, 1993) to provide access to non-visual information for blind and visually-impaired people.

Captioning Format Research and Standards

King and LaSasso (1993-1996) are using *CAP-Media LD* in a series of studies designed to assess the impact of four format features on the comprehension of content and affective response to captions. These features include: (a) methods of indicating speaker identification, (b) methods of indicating non-speech information such as background noise, music, and emotional tone, (c) placement of captions in relation to the video display area, and (d) timing, length, and duration of captions. These studies are part of a project, funded by the U.S. Office of Special Education and Rehabilitative Services (OSERS), which is designed to help set new standards for computer-based multimedia and digital television captioning. The results of these studies are also expected to be generalized to today's analog captioning system.

CAP-Media LD and computer-controlled laserdisc technology were selected for this research for several reasons. First, laserdisc technology permits the researchers to have precise control over the segments of video to be shown and permits non-linear access to the video (ensuring that all subjects see exactly the same portion of video and reducing costs associated with counter-balancing the order of video clips). Second, computer-based captioning is more flexible than Line21-analog captioning (e.g., more fonts and variations, such as placing the captions completely separate from the video, are possible). Third, computer-based captions are less costly and can be of higher quality than captions created in a television editing suite (when the captions involve features--such as different fonts--not possible with current Line21-analog captions).

This line of research grows out of a long history of research on the format features of captioning (e.g., Belduc, 1983; Braverman, 1981; Harkins, 1993-1994; King & LaSasso, 1992-1993). Recent research efforts have been motivated largely by the Television Decoder Circuitry Act of 1990, which mandated that, effective July 1, 1993, every television (13 inches or larger) manufactured in or imported to the United States must have built-in captioning decoder circuitry (DuBow, 1991) and by the work of the FCC and Electronic Industries Association (e.g., EIA, 1992, Hutchins, 1993) to set new technical standards for television captioning.

King and LaSasso (1992-1993) surveyed deaf and hard-of-hearing consumers to determine their opinions concerning captioning format and conducted video preference tests to determine their reactions to proposed new captioning features. Among the results were indications that caption consumers prefer (a) sans serif over serif fonts, (b) captions centered at the bottom of the screen rather than captions placed left-center-right (depending on where the speaker was on the screen), and (c) moderate movement of captions to avoid covering on-screen titles (e.g., names of people and sports scores). Consumers expressed strong preference for the existing Line21 monospace font and black background, when compared to character-generated proportional fonts and varied backgrounds. Whether such preferences were due to familiarity, resistance to change, and/or subtle differences in the quality of the Line21 and character-generated captions could not be determined. Consumers also indicated a desire for choice in features such as font style, font size, and color of the background box (e.g., LaSasso & King, submitted for publication).

In response to an open-ended question about captioning, caption consumers identified problems related to the amount of captioning, accuracy of captioning, obstruction of on-screen titles, and format features such as speaker identification. Regarding speaker identification, some wanted color used for different speakers (as is currently done in the Australian captioning system), some requested that the names of speakers be included with captions, and others wanted different emotional tones to be represented.

Effects of Color-Coding and Placement on Comprehension

Based on the reports of consumer preferences (e.g., LaSasso & King, submitted for publication) and the EIA-608 Report (EIA, 1992), which indicated that caption providers were exploring the use of color for speaker identification, a decision was made to conduct an experimental study of the effects of color and placement (used for speaker identification). [Note: the primary method used in current Line21-analog captioning for television is placement of the captions via eight-tab settings. For on-screen speakers, captions are placed in the location closest to the speaker. For off-screen speakers, captions are placed either in the same location as the last caption for that speaker or in a new location, if the speaker has moved.]

Method

Subjects. Seventy-two students in the Preparatory Studies Program (PSP) at Gallaudet University participated in this study. PSP enrolls deaf and hard-of-hearing students who are not quite ready for college and provides them with studies to improve English and mathematics skills, as well as other college-preparatory studies. Most of these students (82.6%) are profoundly deaf (90+ decibel loss in the better ear), with 5.8% with severe hearing losses (76-89 db), and 11.6% with moderate hearing losses (51-75 db). More than half (56.9%) are female. Only students with normal vision, with or without glasses, are included in the final sample (80 students participated in study, 8 were eliminated due to vision problems or additional handicaps).

Students were randomly assigned to one of four study conditions (described below). Analyses of variance indicate that there are no significant differences between the groups on (a) age, $F(3,68)=.91$, $p=.441$, $MS_e=6.75$; (b) hearing loss (measured in decibels), $F(3,65)=1.98$, $p=.126$, $MS_e=165.04$; or (c) reading level (measured by scores on the *Degrees of Reading Power* test, $F(3,64)=.81$, $p=.494$, $MS_e=54.95$.

Materials. A 15-minute segment from the beginning of Disney's *Sword in the Stone* was used in this study. Digital captions for the video segment were generated, using a 20-point, proportional, sans serif font and a full-screen display (the caption area covered the bottom of the video). Four versions were prepared: (a) captions color-coded for speaker identification, centered at the bottom of the screen, (b) black-and-white captions, centered at the bottom of the screen, (c) color-coded captions with placement dependent on the location of the speaker, and (d) black-and-white captions with placement dependent on the location of the speaker. The last two conditions emulated the system used by the caption provider who had prepared Line21-analog captions for the movie (i.e., captions were placed in the same place as would occur in television captioning).

The test materials for the study involved four separate *Toolbook* applications, using the digital captions (generated by *CAP-Media LD*) for the four conditions. The instructions, controls for playing each video clip, the test question pages, and the captions themselves were identical in all four applications--only the color and placement of the captions varied.

The 15-minute segment was divided into 22 video clips, with stopping points located at phrases where the identity of the speaker might be unclear to someone who could not hear the voices of the characters. Twenty of the stopping points--the actual test items--were selected from a larger set of potential items in a pilot-test conducted with hearing children and deaf and hearing adults. The other two stopping points were for a sample test question to ensure subjects understood the task and a final video clip (without a test question) to complete the video segment. Questions were in the form, "Who said ..." followed by the caption on the screen at the end of the preceding video clip. Answers were in the form of still-frames of the characters from the video, identified as A, B, or C.

Procedures. As previously noted, subjects were randomly assigned to one of the four conditions. Testing took place over three days in a given week, with each subject participating in a single test session of approximately 45 minutes on one day. Subjects watched the video on a 27" computer monitor in groups of 2-10 and recorded their answers in paper-based test booklets. Following the test, all subjects were asked to respond to a set of Likert-type items concerning their opinions of the captions they saw, and subjects in color-coded conditions were asked to name the color used for each character (black-and-white versions of still-frames from the video were used as prompts). The five-point, Likert-type scale had choices of "I like it. I like it a lot. I have no opinion. I don't like it. I don't like it a lot," with "it" being the feature of interest.

A factorial analysis of variance was conducted with test scores as the dependent variable and color and placement as independent variables. Chi-Square statistics, using an experiment-wise alpha level of .01, were calculated to identify those test items that best discriminated among the groups. Descriptive statistics and Chi-Square statistics were also used for the Likert-type items on affective responses to the digital captioning. A Pearson Product-Moment Correlation was calculated between test scores and scores on the five-item test identifying the color for different characters (for only those subjects who saw color-coded captions).

Results

Table 1 contains descriptive statistics for test scores (maximum score of 20) for the four conditions. There are significant differences between those who saw color-coded captions and those who saw black-and-white captions, $F(1,68)=12.55$, $p=.001$, $MS_e=14.36$. There are no significant differences for (a) placement, $F(1,68)=.34$, $p=.561$, $MS_e=14.36$, or (b) the color-by-placement interaction, $F(1,68)=1.89$, $p=.174$, $MS_e=27.09$.

Table 1
Mean Test Scores, Standard Deviations, and Number of Subjects for Study Conditions

	Centered	Line21 format	COMBINED MEANS
Color-Coded	16.29 (3.60) ($n=17$)	17.00 (3.22) ($n=18$)	16.65
Black-and-White	14.35 (3.98) ($n=17$)	12.60 (4.22) ($n=20$)	13.48
COMBINED MEANS	15.32	14.68	14.99

There are significant group differences in the response patterns of subjects for only two of the test questions, #8, $\chi^2(6,N=72)=18.65, p=.0048$; #16, $\chi^2(6,N=72)=24.13, p=.00049$. Both test items relate to video scenes where the speaker is off-screen and the caption is located near one of the on-screen characters. In the test item, the on-screen character is a distractor (incorrect choice). On item #8, 65% of those who saw black-and-white, Line21 format captions selected that incorrect choice (compared to 41% of those who saw black-and-white, centered captions and only 12% of those who saw color-coded captions). On item #16, 40% and 47% of the subjects who saw black-and-white captions (Line21 format and centered, respectively) selected the on-screen character (compared to 14% of those who saw color-coded captions).

There are no differences among the four groups on their overall affective responses to digital captioning. For this analysis, frequencies for "like and like a lot" are collapsed, as are those for "don't like and don't like a lot." Approximately three-fourths of the subjects like the captioning (78%), font style (78%), font size (74%), and the mixed, upper- and lower-case of the lettering (80%). Of those who saw color-coded captions, 79% like them. For placement of captions, 78% of those who saw centered captions like centering, and 71% of those who saw Line21-format captions like variable placement. The percentage of subjects who had no opinion concerning each feature of the captions ranged from 12% to 21%, and the percentage of subjects who did not like the feature ranged from 3% to 14%.

Approximately two-thirds (67%) of the subjects who saw color-coded captions could name the colors of the captions for all five of the main characters. A moderate, positive correlation exists between test scores and the number of colors correctly identified, $r=.352, p=.035, n=36$.

Discussion

Results of this study show that color-coding has a significant effect on comprehension, when comprehension is narrowly defined as understanding of speaker identification for potentially-confusing items. These results, however, cannot be interpreted to mean that color-coding should replace the current line21 format that uses placement for speaker identification. First, color-coding is unusable for that portion of the population that is color-blind (thus, redundancy in any coding system that uses color will be necessary). Second, color-coding is unwieldy when the number of main characters is large. Third, complex color-coding systems may not be usable (e.g., The Australian Caption Centre recently reduced the complexity of their color-coding system, based on viewer feedback; C. Grimmer, personal communication, December 15, 1993).

Color-coding for speaker identification, in some combination with placement and inclusion of character names in the captions for off-screen speakers, should be studied further. When there are a limited number of main characters in the story, color-coding, as shown in the present study, appears to increase comprehension. The next study by King and LaSasso (1993-1996) will address how and how well children learn, over time, a color-coding system used for speaker identification.

Summary

The study reported in this paper is the first in a series of studies on captioning format for multimedia and digital television. The results have application for both captioning environments (for caption providers) and screen design (for multimedia developers). Most important, however, is the recognition that--without captioning--deaf and hard-of-hearing people do not have independent access to audio-only information. Tools like *CAP-Media LD* are essential, if this important audience is to have full and equal access to digital media.

References

Armon, C., Glisson, D., & Goldberg, L. (1992). How closed captioning in the U.S. today can become the advanced television captioning system of tomorrow. *SMPTE Journal, 101*, 495-498.

Beldue, I. P. (1983). Fixed versus variable letter width for televised text. *Human Factors, 25*(3), 273-277.

Bess, L. P. (1993, September/October). Closed-Caption TV: A new era. *SHHH Journal, 14*(5), 8-10.

Braverman, B. (1981). Effects of language level and visual display on learning from captioning instruction. *Educational Communication and Technology, 29*(3), 147-154.

DuBow, S. (1991). Television Decoder Circuitry Act: TV for all. *Temple Law Review, 64*, 605-614.

Electronic Industries Association (EIA). (1992). *Draft standard: EIA-608: Line 21 data services for NTSC.* Chicago: Author.

Harkins, J. E. (1993-1994). *Caption features for indicating non-speech information: Research toward standardization.* Grant funded by the US Department of Education, Office of Special Education and Rehabilitative Services.

Hutchins, J. (1993, September/October). The captioning technology of the future is being developed today. *SHHH Journal, 14*(5), 22-24.

Institute for Academic Technology. (1992, Fall). Faculty develop programs for listening comprehension and text synchronization. *IAT Briefings: The Institute for Academic Technology*, pp. 14-15.

Jordan, I. K. (1993). A matter of access: Technology for deaf and hard-of-hearing people. *TechNos, 2*(2), 26-28.

Kaplan, D., & De Witt, J. (1994). *Telecommunications and persons with disabilities: Building the framework. The second report of the Blue Ribbon Panel on National Telecommunications Policy*. Oakland, CA: World Institute on Disability.

King, C. M. (1993). Multimedia captioning: Synchronized text and audio presentations. In H. Maurer (Ed.), *Educational Multimedia and Hypermedia Annual, 1993* (pp. 609-614). Charlottesville, VA: Association for the Advancement of Computing in Education.

King, C. M., & LaSasso, C. J. (1992-1993). *Acceptability, readability, and comprehensibility of caption format features: Perspectives and responses of deaf and hard-of-hearing caption consumers.* Grant funded by the US Department of Education, Office of Special Education and Rehabilitative Services.

King, C. M., & LaSasso, C. J. (1993-1996). *CAP-MEDIA: Captioning for multimedia and digital television.* Grant funded by the US Department of Education, Office of Special Education and Rehabilitative Services.

LaSasso, C. J., & King, C. M. (submitted for publication). *Deaf consumers' views about captioning.*

National Captioning Institute. (1991). *An international bibliography of captioning and subtitling literature.* Falls Church, VA: Author.

Short, D. D. (1992). Instruction in a postliterate culture: Multimedia in higher education [Invited article]. *Journal of Information Systems Education, 4*(3), 11-15.

Short, D. D. [Developer], & King, C. M. [Design Consultant]. (1994). *CAP-Media LD: Synchronized text for laserdiscs* [computer program]. Durham, NC: Institute for Academic Technology.

WGBH Educational Foundation. (93). *Descriptive video services.* Boston: Author. [125 Western Avenue, Boston, MA 02134]

Acknowledgments

Support for the experimental research described in this paper was provided in part by grants from the U.S. Department of Education, Office of Special Education and Rehabilitative Services: Educational Media Research, Distribution, and Training Program (Grant Nos. H026R10002 and H026R30011), with additional funding from Gallaudet University. Opinions and views expressed in this paper are those of the authors and do not necessarily reflect the positions of any government agency or the university.

The Institute for Academic Technology (IAT) is a partnership between IBM and the University of North Carolina at Chapel Hill on behalf of education. For information about its courses, programs, and services, contact the Institute for Academic Technology, 2525 Meridian Parkway, Suite 400, Durham, NC 27713, (919) 560-5031, e-mail: info.iat@mhs.unc.edu.

A Hypermedia-Based Educational System With Knowledge-Based Guidance

ANDRE KUSHNIRUK
Center for Medical Education
McGill University, 1110 Pine Avenue West,
Montreal, Quebec, Canada, H3A 1A3
email: cx2v@MusicA.McGill.ca

HUAIQING WANG
Department of Computer Science
University of Toronto, 6 King's College Road, Room 283E
Toronto, Ontario, Canada, M5S 1A4

Abstract: This paper describes an educational system that integrates hypermedia with knowledge-based technologies. The system contains a hypermedia component that allows students to browse through information contained in a domain knowledge base dealing with urban planning and land use. The hypermedia system is coupled with an intelligent tutoring component that allows students to attempt exercises. User "views" of the hypermedia network are dynamically generated, based on the system's analysis of student problems, misconceptions and learning needs. A representational scheme is described that integrates the components of the system into a coherent object-oriented framework. It is argued that a combination of knowledge-based techniques with hypermedia technology will be important in the development of systems that are flexible and adaptable enough to allow for the full potential of hypermedia in education.

1. Introduction

Hypermedia systems allow for the access of large quantities of information in flexible and interactive ways, facilitating the user's exploration of knowledge and learning. In hypermedia systems, users can navigate and browse through linked networks of information nodes consisting of varied media (e.g. text, images etc.). The potential of such systems is considerable and with the growing popularity of interactive graphical user interfaces and technologies such as multimedia, the need for techniques that allow users to access and navigate through large amounts of information will continue to grow. In the domain of education, hypermedia holds considerable promise. However, some basic problems still remain to be resolved. The flexibility that such environments provide the student can greatly facilitate learning and the exploration of knowledge on the part of the student. On the other hand this flexibility can also lead to common problems of disorientation. Thus users of hypermedia systems may find themselves "lost in hyperspace", not knowing where they are, what information is relevant to their current level of understanding and learning needs, and finally, how one can get to that information.

The need for intelligent guidance for users of hypermedia systems has been noted by a number of authors (David, Thiery & Crehange, 1989; Woodhead, 1991). Possibilities include intelligent means to alleviate information overload and reduce the complexity of hypermedia information presented to users to more manageable and meaningful "views". This could be extended to providing such intelligent presentation of information in ways that are tailored to the information needs and level of expertise of particular users. Key to the success of such systems will be the integration of hypermedia with knowledge-based technologies, such as expert systems (Briggs, Tompsett & Oates, 1991; Brusilovsky & Zyrajanov, 1993). A continuum of possible architectures that integrate hypermedia with knowledge-based systems can be conceptualized, ranging from loose coupling where the system architecture consists of separate knowledge-based and hypermedia modules, to tightly coupled architectures, consisting of integrated knowledge-based and hypermedia elements that work in unison, sharing data, knowledge representations and allowing cooperative reasoning (Ragusa, 1994). For example, Wang and Kushniruk (1992) describe a tightly coupled architecture that consists of a hypertext knowledge base that students can browse through, with an embedded intelligent-tutoring system (ITS) component.

In this paper it is argued that rigidly structured hypermedia networks - i.e. networks where all linkages between individual information nodes are "hardwired" or fixed by the system designer will not allow the user sufficient flexibility that will be needed for accessing the increasingly larger educational knowledge bases that are currently being developed. It is further argued that what is needed are schemes and knowledge representations that will allow for both static (fixed) and dynamically generated linkages among information nodes in the hypermedia network. Such dynamic linkage of information contained in a hypermedia network could be based on the student's learning performance and demonstrated level of expertise, resulting in a greater degree of adaptability in educational hypermedia systems. The display of information, the level of detail provided, the highlighting of information and the creation of possible paths (or user "views") through the hypermedia network that the user could access can be based on information contained in and continuously updated in a system model of the student. In previous work (Wang & Kushniruk, 1993), the authors have succeeded in using a student model to select exercises appropriate to the level of the student's learning. In this paper we extend this research to the design of a tightly coupled ITS-Hypermedia educational system. Various forms of information, including hypertext, graphic images, and test questions, can be accessed by students, while a knowledge based component continually monitors student performance and provides intelligent guidance in the presentation of the hypermedia. This paper explores some of these ideas in the context of an example application in the domain of teaching geographical concepts.

2. Overall System Architecture

This section provides a description of the overall architecture of a hypermedia-based educational system for teaching principles of urban planning, known as the "LAND USE Tutor". From figure 1 it can be seen that the system consists of a number of components. The four main components are the following: 1. a domain knowledge base (KB-Domain) 2. a user knowledge base (KB-User) 3. a knowledge base for controlling the hypermedia presentation (KB-Control), and 4. a display module.

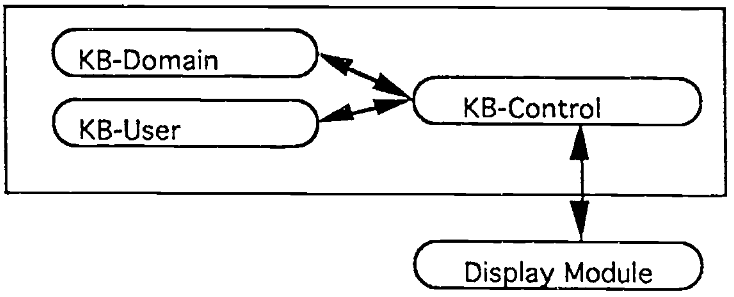


Figure 1. System Architecture

The KB-Domain contains domain specific information, for example information about site selection in planning urban developments. This information is accessed during the student's browsing through the system and consists of information nodes (using an object-oriented knowledge representation scheme described below) in a hypermedia network. The KB-User contains information about the user's interaction with the system and consists of a user model, that is developed by the system, as result of the student's browsing history and performance on exercises. The KB-Control is responsible for providing intelligent guidance by selecting a set of dynamically generated paths through the hypermedia network that the student can access. After every user action (e g. mouse click, selection of hypertext etc.), one or more objects representing the action are created and asserted into working memory and a forward chaining inference engine is invoked to set one or more dynamic paths. The inference engine can apply knowledge contained in the KB-Domain and KB-User (e.g. information about the user's previous interactions with the system and difficulties) in conjunction with information about the user's most recent action. As a result, certain parts of the hypermedia network contained in the KB-Domain may be made accessible and salient (through the creation of dynamically generated links). The KB-Control also interacts with the Display Module, sending it selected parts, or "views", of the hypermedia network to be displayed.

3. Knowledge Representation and System Operation

Knowledge contained within the LAND USE Tutor is represented, using the object-oriented paradigm, in classes, subclasses and their instances, which are organized into instance hierarchies. In this paper we will describe some examples of classes using C++ format. For example, the hypermedia network is represented using a class hierarchy (a part of which is shown in figure 2) with class Geographical_HyperMedia (a subclass of System_Object) having the following description:

```
class Geographical_Hypermedia : public System_Object
        {
         String title;
         SET <Geographical_HyperMedia> Static_links;
         SET <Geographical_HyperMedia> Dynamic_links;
         ...}
```

This description indicates that all hypermedia information nodes in the system's domain knowledge base (belonging to class Geographical_HyperMedia) have a title (of type String), and two sets of links to other hypermedia nodes: one set of static links (that do not change) to other nodes and another set of dynamic links to other nodes (that can be determined at run-time, as a result of the student's interaction with the system).

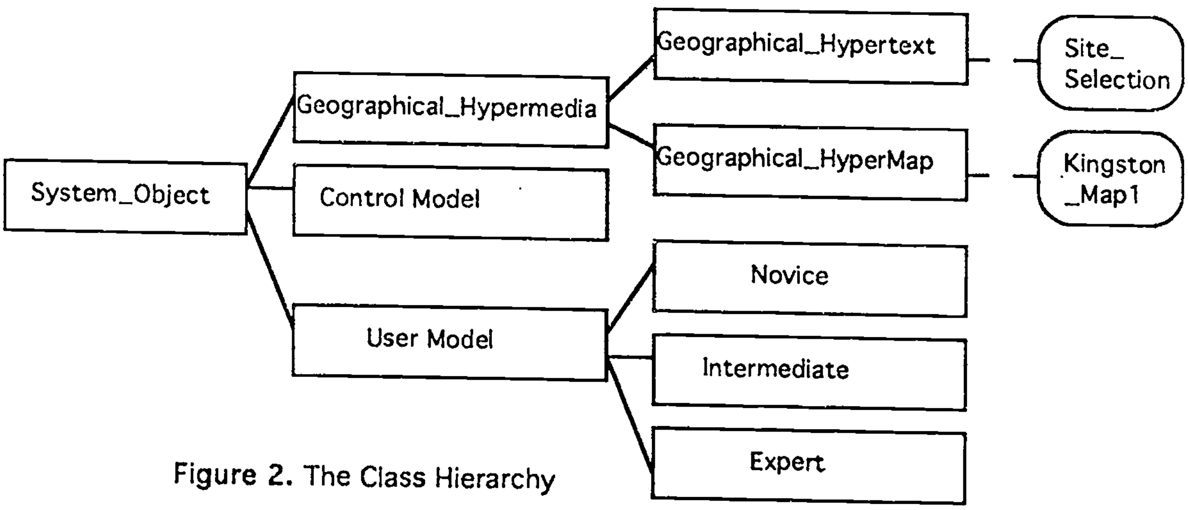


Figure 2. The Class Hierarchy

As can be seen in figure 2, two subclasses of the class Geographical_HyperMedia are the classes Geographical_HyperText (for hypertext information) and Geographical_HyperMap (for graphic information). A partial description of these classes is given below:

```
class Geographical_HyperText : public Geographical_HyperMedia
    {
    TEXT hypertext;
     ... }
class Geographical_HyperMap : public Geographical_HyperMedia
    {
    BitMap map;
    SIZE size;
    ... }
```

In the system, a particular instance of geographical information (e.g. an instance of a hypermedia node containing information about choosing site locations for urban development, such as the node "Site_Selection" in figure 2), will have only static fixed links and no dynamic links to other information nodes at the beginning of a session with a student (i.e. the list of dynamic nodes associated with that node is initially set to NULL). However, based on the results of the student's performance on exercises (see the next section on Student-Tutor Interaction for an example), dynamic links to other parts of the hypermedia network may be established (thereby adding new paths through the network that the student has access to in browsing). The description of a particular instance of geographical hypertext information, containing information about site selection ("Site_Selection"), is given below (note that this instance has a fixed, or static, link to an instance containing information about topography) :

```
Site_Selection InstanceOf Geographical_Hypertext
  {
    title: "Site_Selection";
    text: "Many factors must be considered when selecting
           a possible site for building or land development.
           These factors include:
                  1. {\hypertext Topography}
                  2. ... ";
    Static_links: {Topography, ...};
    Dynamic_links: NULL;
    chapter: Chapter_3;
    keywords: { "land use", "urban planning", ... };
    }
```

The ITS component of the Tutor is capable of detecting student misconceptions in problem solving (the design of this component is described in Wang & Kushniruk, 1992 and Wang & Kushniruk, 1993). This determination of possible student problems and misconceptions is based on a knowledge-based analysis of the student's responses to exercises. The dynamic linking of hypermedia nodes, based on the existence of student misconceptions, is triggered by the firing of the following generic rule:

```
FORALL $x InstanceOf Geographical_HyperMedia
   IF (established existence of misconception $m)
          AND Find_Related_HyperMedia $m $y
THEN  $x.AddDynamicLink($y)
```

This rule indicates that for a particular instance (instantiated to $x) of a geographical hypermedia node (e.g. containing information about site selection), if a related student misconception (instantiated to $m) is detected and if another hypermedia node (instantiated to $y) is found that contains remedial information related to the misconception, then the two information nodes (i.e. $x and $y) will be dynamically linked. Thus the two nodes will be directly connected at run time in the hypermedia network.

4. Student-Tutor Interaction

When the student first invokes the LAND USE Tutor, information about the student's previous interaction with the Tutor is loaded. A window containing a list of factors to be considered in selecting geographical sites for development is initially displayed from which the student can select. For example, if the student selects information about Topography, this results in the display of another hypermedia window containing hypertext information about the role of topography in selecting a site for building development (keywords in the windows are highlighted for which further information can be obtained). At any point in the student's browsing through the hypermedia network, the student can decide to test his/her knowledge of the information he or she is reviewing by selecting the exercises option from the buttons that appear at the top of each text window. For example, in Figure 3 the student has selected the exercise option while reviewing information about topography. At this point, the Tutor will select an exercise for the student to attempt, based on the system's assessment of the student's level of competence (see Kushniruk & Wang, 1991, and Wang & Kushniruk, 1992 for a discussion of exercise selection based on a student model). In the example interaction, the exercise for the student to attempt is presented in the upper right-hand window in figure 3.

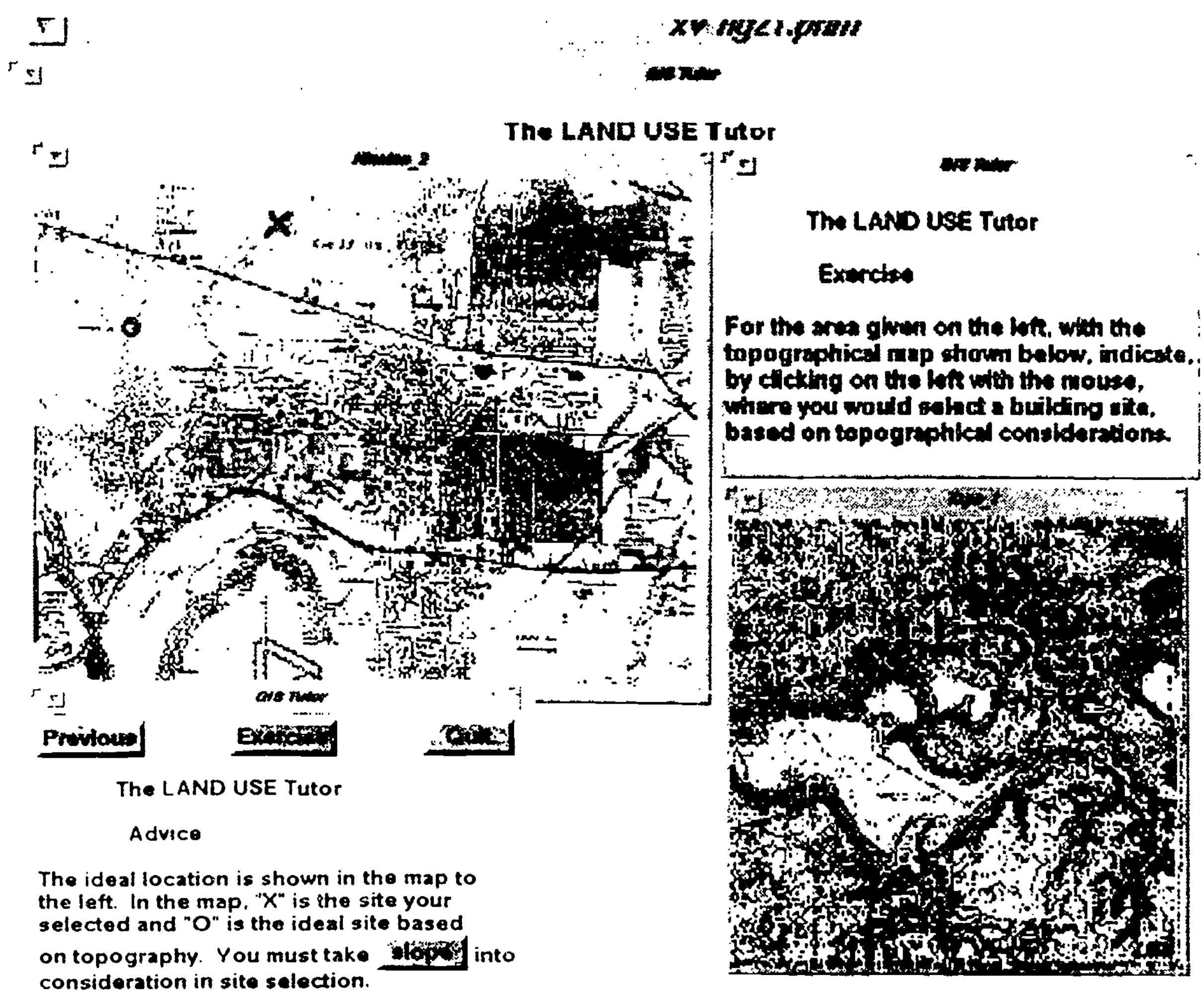


Figure 3. Example of Student-Tutor Interaction

For the exercise in the example, the student is instructed to indicate on the map given in the upper left-hand side of figure 3, where he/she would select a site for a building development. In addition, a topographical map is presented for reference purposes to the student (the map in the lower right-hand side of figure 3). The student selects the site area by clicking the mouse on the desired area of the map in the upper left-hand side of the figure. A knowledge-based expert system component is then invoked to analyze the student's response (i.e. the student's indication of where to locate the site on the map). A set of rules, associated with this particular instance of a student exercise, is invoked to determine if the site indicated by the student matches the ideal location for this particular exercise (i.e. the correct location). If it does not, a rule-based analysis of the student's answer is then invoked (using the techniques described in Wang & Kushniruk, 1993) and hypotheses as to the nature of any underlying student misconceptions are asserted in the Tutor's knowledge base.

The assertion of a hypothesis regarding the existence of a student misconception or misunderstanding may result in the generation of a dynamic hypermedia link to other nodes containing remedial or related information (thus directly linking the current node containing the exercise with other parts of the hypermedia network). In the example, the student had incorrectly located the building site, that is, without proper regard for the slope of the selected site. Upon analysis of the student's answer, an assertion was made that a misconception exits regarding the student's understanding of the role of slope in selecting a site based on topography. Using the generalized rule described in section 3 above (for dynamically generating links between parts of the hypermedia network), a link is made to the part of the domain knowledge base containing an information node about the concept of slope. Thus, in the window that appears in the lower left-hand corner of figure 3, advice is provided to the student, indicating that he/she must take slope into account. Note that the keyword "slope" has now been dynamically associated with information regarding that concept and it is highlighted as being selectable in the window containing advice in figure 3 (i.e. the student may now directly access information about the concept of slope by clicking on that keyword). Thus links within the hypermedia network can be dynamically altered as a result of the student's performance on an exercise and the resulting analysis of student errors.

5. Conclusions

In this paper we have described an educational system that integrates hypermedia with knowledge-based technology, providing user "views" of the hypermedia network based on an analysis of the student's problems and learning needs. It has been shown in this paper that knowledge-based techniques and representations can provide the needed means to achieve a degree of adaptability in hypermedia systems that is both practical and useful. In summary, some of the features of the system described in this paper include the following:

- The LAND USE Tutor integrates both hypermedia and knowledge-based technologies.
- The system incorporates a knowledge representation scheme that allows for the representation of the many types of knowledge required in such a hybrid hypermedia/knowledge-based system, within a coherent and integrated object-oriented framework.
- Knowledge contained within the Tutor is organized as a number of knowledge bases, simplifying the system's operation and the updating and acquisition of new knowledge.
- The LAND USE Tutor incorporates an ITS component that allows the student to attempt exercises related to topics being browsed. This component is also responsible for determining if underlying student problems or misconceptions exist.
- Based on the results of diagnostic processing, links between information nodes in the hypermedia knowledge base can be dynamically generated or modified, adding a level of system adaptability.

In conclusion, we feel that many of the ideas and techniques described in this paper could be extended to the design of a variety of types of computer systems. Furthermore, we feel that the integration of hypermedia and knowledge-based technologies will become increasingly important in the development of hypermedia applications and will be key in the development of future advanced educational systems.

6. References

Briggs, J., Tompsett, C. & Oates, N. (1991). Guiding learners through a medical hypertext database using knowledge based rules. In R. Bottino, P. Forcheri and M. Molfino (Eds.) *Proceedings of the Sixth International PEG Conference*, Genova, 571-577.

Brusilovsky, P., Zyryanov, M. (1992) Intelligent tutor, environment and manual for physical geography. *Proceedings of the Seventh International PEG Conference*, Edinburgh, Scotland, 63-73.

David, A., Thiery, O., Crehange, M. (1989) Intelligent hypermedia in education. In H. Maurer (Ed.) Computer Assisted Learning. *Proceedings of the 2nd International Conference ICCAL'89*. Springer-Verlag: Berlin, 53-64.

Kushniruk, A., Wang, H. (1991). An expert system for providing help to computer users. *Proceedings of the World Congress on Expert Systems*, Orlando, Florida, 1469-1477.

Ragusa, J. (1994). Models of Multimedia, hypermedia and 'intellimedia' integration with expert systems. *Proceedings of the Second World Congress on Expert Systems*, Lisbon, Portugal.

Wang, H., Kushniruk, A. (1992). The UNIX tutor. In C. Frasson, G. Gauthier, G. McCalla (Eds.) Intelligent Tutoring Systems. *Proceedings of the Second International Conference on Intelligent Tutoring Systems, ITS'92*. Springer-Verlag: Berlin, 317-324.

Wang, H., Kushniruk, A. (1993). An intelligent tutoring system for UNIX. *Proceedings of the Seventh International PEG Conference, Edinburgh*, Scotland, 261-269.

Woodhead, N. (1991) *Hypertext and Hypermedia*. New York: Addison-Wesley Publishing Company.

MUSLI*
A MUlti-Sensory Language Interface

J. LENNON
Department of Computer Science
University of Auckland, New Zealand

H. MAURER
Institute for Information Processing and Computer Supported New Media (IICM)
Graz University of Technology, Graz, Austria
and
Hypermedia Unit, University of Auckland, New Zealand

Abstract: While the expansion of worldwide hypermedia systems has opened new gateways to an undreamt-of commerce in information, this has led to valuable material being archived faster than it can possibly be read. In this paper we suggest, as a new and vitally important area of research, the development of multimedia tools that allow more efficient and effective "reading" and "writing" using new types of hypermedia documents. We look beyond traditional hypertext, taking the first steps towards a language of dynamic abstract symbols that will help authors to express precisely and concisely both concrete and abstract ideas. MUSLI readers will enjoy searching for detailed information by invoking new multimedia "speed reading" techniques.

1. Introduction

In spite of, and partly because of, dramatic advances in information technology, keeping up to date in any discipline is now almost an impossibility. We know that hypertext systems are available which offer certain solutions to the problem of efficient information retrieval (Maurer, Kappe, Scherbakov, & Srinivasan, 1993; Nielsen, 1990). However we need more help than systems such as these are currently providing.

Although Socrates in Plato's *Phaedrus* argued that writing a language destroys it, reading text is certainly a faster method for gaining information than listening to an orator! Yet we now have a generation of young people many of whom find reading "too slow". People need to be able to scan information as well as good speed readers can scan printed text, or even better. We need to have more control over our communicating media, the level of information we feel motivated to reach, and the pace we wish to set ourselves. Both at leisure and at work we are now accustomed to a very rich proportion of pictorial information in communication; and the importance of visual information must not be underestimated. In particular, certain types of animated pictures may well be worth more than a large number of static ones. Unfortunately we are still severely limited by the lack of tools for producing documents that involve all the senses. One of the aims of MUSLI (MUlti Sensory Language Interface) is to integrate dynamic abstract symbolism ("abstract movies") into hypermedia technology, allowing the efficient expression of ideas difficult to communicate otherwise.

In this paper we use the term "hypermedia" to mean linked multimedia documents embedded in a network. We restrict ourselves to exploring asynchronous communication in hypermedia systems even though

* The Name MUSLI evolved from MUSIL – MUlti Sensory Information Language. The original choice was apt since it is also the name of the famous Austrian poet and novelist Robert Musil (1880-1942), who was not only a master of that abstract communication device, language, but employed language on a notably abstract level in "inner monologues" or "streams of consciousness" – the technique familiar to readers of James Joyce's novels. The new name, MUSLI: A MUlti-Sensory Language Interface, is appropriate in another sense – nutritious cuisine for mind and body!

synchronous communication may well give rise to still more exciting and challenging research. For example, Maurer and Carlson (1992) point out that although our second most important sensory organ, the ear, has a counterpart, the mouth, our eyes have no such counterpart. We cannot project mental images for other people to capture. The authors suggest that hypermedia systems may develop to a stage where we can produce concrete and abstract projections of our mental images, by computer, so easily and naturally that they will provide us with a new dimension in communication capable of transforming our lives. As wide-area and broad-band networks expand we have an increasingly powerful medium for communication and collaboration, whose impact will not be simply to tie information together but, much more significantly, to "tie people together" (Maurer, 1993a). This, coupled with the fact that multimedia machines are becoming increasingly powerful and affordable, implies that we have the mechanics for truly interactive hypermedia communication systems. Tools for generating new types of hypermedia documents must be developed so that we can make more efficient use of information than is currently possible.

In this paper we shall introduce two new ways of "writing" multimedia material in forms that will aid "reading". We shall expand the idea of browsing and take the first steps toward introducing a more flexible system for conveying information – MUSLI.

Humanity has evolved so many ingenious systems of communication since civilisation first dawned that it makes sense to look briefly at a few of these before we propose a new one.

2. Alternative Forms of Communication

One of the most striking features about communication using signs and symbols is the tremendous variety in medium and form – from the simplest runes on clay or parchment to the most intricate designs on human skin. Prehistorians believe that many ancient cave drawings found in France can be considered as forms of writing: they have discovered groups of drawings which taken together form codes (Jean, 1989). Paintings in the Renaissance were richly redolent with emotionally charged symbolism. Carvings on wood, stone and metal range from concrete signs to abstract symbolism. Signalling devices range from smoke signals to systems of bells, mirrors and musical instruments, to semaphore and electric morse-code – and in the future we are bound to discover new options using digitised sound. Cistercian monks who practise vows of silence have developed highly refined systems of gestures (Umiker-Sebeok, & Sebeok, 1987). Many signs used by deaf people today (Jeanes, 1982) are based on these and it is significant that in many respects signing can convey meaning more quickly than spoken language.

3. The Basic Idea

MUSLI combines hypermedia techniques with the novel idea of dynamic abstract symbols. We aim to facilitate communication by developing a system in which ideas can be expressed as clearly, precisely, and artistically as desired while giving readers the deep satisfaction of using a rich and highly meaningful medium they can control. This paper is to be seen as the very first step, an initial contribution that we hope will encourage scientists to work at opening up new realms in presenting, archiving and communicating ideas.

Obviously any new human endeavour that is worthwhile requires an initial investment of effort. Learning any new language demands dedication and since the system we propose includes abstract symbols it is certain to be no exception.

It is a well known fact that good readers are often disappointed when they see the film version of a book they have read. The problem is often due to the fact that while books leave room for personal interpretation and imagination, movies do not. The "beautiful mountain" of the book will "look different" in the minds of different readers, but all viewers of a movie see, inevitably, the same "beautiful" mountain. What has happened is that there has been a shift in the level of abstraction – the abstract notion "beautiful mountain" has been replaced by a concrete version of a mountain. In MUSLI the proportion and level of abstraction can be chosen to fit the situation, varying from abstract to very concrete as the movie producer feels is appropriate to communicate exactly what is desired.

It is an interesting observation that ideas can be thought of as having the form of a mountain (Maurer, 1992) and it is significant that we talk about having a "point of view" on a topic. As an example of looking at something in more than one way consider the well known problem in first year computer science courses of calculating the Fibonacci series. Many books use this example to teach recursion and then almost apologise for mentioning that it can be solved better using iteration[1]. Very few lecturers indeed show a third way of looking at the problem using dynamic programming. Yet each of the three methods provides insights that are valid and complementary views of the problem. In an interactive multimedia system students can compare ideas to an even greater extent than ever before.

In any multimedia presentation readers should certainly be able to stop and start at will, rewind, review, and then resume. The system should also enable readers to master new material by building up additional skills as they go. In a MUSLI movie (i.e., a hypermedia document that includes special moving abstract symbols) users can stop and display additional information – at any time they feel ready for it. As explained in Section 7, any group of symbols can be condensed to a single "short-form", and double-clicking on the composite symbol will decompose it again. This gives the user significant control over the amount of data on the screen at any particular time. The user will be able to obtain additional detailed information about scenes, characters, ideas, etc. when it is relevant to them, as well as readily refreshing their memory if they have forgotten the meaning of a symbol. This reference facility in itself will give MUSLI documents significant advantages over traditional books.

More significantly, users will be able to skim condensed versions of a document, searching for relevant sections, before decomposing the symbols to display pertinent information. With facilities such as these, it will be possible to speed-read MUSLI documents faster than printed text.

As systems become available that include electronic personal assistants (Cypher, 1993; Maurer, 1993b; Zissos, & Witten, 1985) readers of MUSLI documents will obtain further help in overcoming the widespread problem of information overload. The system will "learn" what level of detail needs to be displayed so that the "reader" can access needed information more efficiently.

4. What MUSLI is Not

The system we propose must not be misunderstood. It enhances traditional multimedia documents. We propose to extend communication, not restrict it! MUSLI is not simply a system of cartoons – cartoons are designed to be understood without any investment in learning and are quite close to a simplified reality. MUSLI is not a silent movie or some system of signs. It will make appropriate use of computer graphics (2D and 3D), cartoons, audio clips (music and speech both analogue and digital), video clips (analogue and digital), photographs (digitised), animation including 3D modelling; and of course text. Naturally there will still be movies – we delight in the work on human animation being done by scientists and artists such as those at the MIRALab at the University of Geneva (Magnenat-Thalmann, & Thalmann, 1991). Visualisation will continue to be an important means of communication (IEEE, 1993) and we watch with interest the work being done in the field of virtual reality. It is our hope that all of these achievements will be eventually integrated into a single grand design: MUSLI.

5. Using Sound, Speech and Text

Sound can frequently be used to good effect. There is hardly a better way of setting moods for scenes and indicating such things as a really good thunderstorm than by using appropriate sounds. An interesting point made by Alty (1993), is that peripheral or parallel streams of information containing redundant information can be of importance in helping students understand complex ideas.

Although speech is inherently slow, there may be alternative ways of processing it. We note, for example, that it is now possible for computers to play back speech at twice the recorded speed and still maintain an acceptable level of pitch and clarity.

One interesting idea, that should be considered further, is to extend the use made of digitised speech. For example, for scanning purposes, users could display on the screen the waveform patterns of the speech. It may be possible to search for particular sections to play by learning to "read" waveforms, at least in a broad sense; for example, changes in volume may indicate passages of significance. If the wave patterns are expanded and the

[1] Some warn of arithmetic overflows though what really overflows is time, not size! If m = f(n) is the n'th Fibonacci number, then the recursive function: function f(n integer) integer, if(n=1) or (n=2) then f := 1 else f := f(n-1) + f(n-2); takes O(m) steps to compute. Since $m = O(c^n)$ with c~1.4 it takes exponential time to compute f(n). The size of f(n) measured in the number of digits, however, grows only linearly!

amplitude is displayed in, say, red while the pitch is displayed in blue, additional information can be gained. In particular, it should be possible to search digitised speech more efficiently than at present. For example, alternating passages of high and low pitch may well mark dialogue. Exciting passages will stand out. Sentences will be seen marked off from one another and other information will be visible such as the occurrence of a question. If the author of the document provides signposts in the form of icons alongside key passages the users will be further aided in their searches. It should also be possible for specialised computer programs to scan the material in greater detail. With methods such as these authors will be able to produce documents allowing "readers" to understand ideas more readily.

The possibility of incorporating existing systems of signs and gestures is an interesting alternative to both speech and text and needs further research (Dreyfuss, 1972).

6. A First Tentative Proposal

In our move towards a theory of dynamic abstract symbolism, we explored research in semiotics in an attempt to choose a grammar. Several alternatives such as the grammar of Conceptual Dependency Theory (Schank, 1975; Schank, & Abelson, 1977) remain to be explored. However as an anchor point for our first proposal we quote from *An Introduction to Functional Grammar* (Halliday, 1985): "Our most powerful conception of reality is that it consists of 'goings-on': of doing, happening, feeling, being." Halliday identifies the three most common types of processes: material processes, mental processes and relational processes. These categories seem amenable to representation. We shall begin by symbolising examples of the three processes, considering each process in turn.

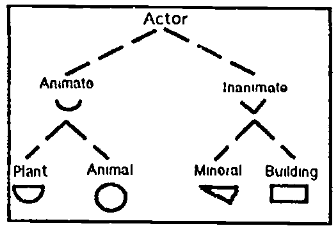
1. Material processes: processes of doing. Here we have the common "Actor, Process and Goal" construct that suggests animation. We can animate actions for walk, run, jump, dance, break, make, give, take, etc.
2. Mental processes: processes of sensing. Since these can be regarded as directed actions we shall indicate them by dynamic lines or arrows. For example, love can be a golden shaft directed towards the object. Liking will be represented by a less intense yellow while hate will be brown or black. Directed fear and worry will be jagged shafts.
3. Relational processes: processes of being. Here we shall consider two classes: those that are attributes of the object and those which can be considered as states of the object. Attributes we shall assign as associated symbols as described in Section 7. States we can indicate by colouring objects, for example, pink for happy and blue for sad.

7. Using Abstract Symbols

The basic symbols we introduce will need to be simple – yet be beautiful in their own right. The more frequently a symbol is used the simpler it should be. Additional characteristics will be added by association and only when needed. Obviously we shall not have to introduce new symbols for all words: we shall include accepted signs, (Liungman, 1991), such as an X for "forbidden", and we can have dictionaries of known shapes and symbols to call on. As noted in Section 3 we shall try to avoid using shapes that are too lifelike, partly because this limits the reader's imagination and partly because using abstract symbols gives us more freedom to add attributes without contravening our built-in sense of appropriateness. For example, while it may be quite inappropriate to paint a life-like symbol of a beautiful woman a golden yellow colour, a golden circle definitely transmits the idea of beauty.

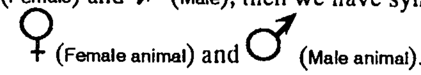
There are two distinct facets to abstraction: abstraction of appearance and abstraction of senses (feeling, thinking and emotion). As described above, a golden circle may represent beauty of appearance. But beauty applies just as significantly to the abstract, and it is qualities such as feelings, thoughts and emotions that have proved difficult for existing visual forms of communication. Abstract symbols can take on additional attributes with no conflict. Beautiful feelings, beautiful thoughts, beautiful emotions: all can be abstractly displayed as golden states. Even speech (or thought) balloons which look clumsy on lifelike characters may be artistically attached to symbols.

We shall begin by introducing basic symbols using the following classification system:

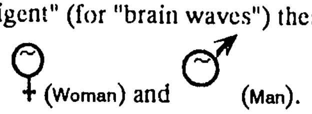


7.1 The Addition of Attributes

We shall be able to modify basic symbols by associating attributes. These modifiers may be either one or more associated symbols, or an integral part of the basic symbol, such as colour. For example, if we introduce the common biological symbols: + (Female) and ↗ (Male), then we have symbolic representations for:



Using the symbol ~ to represent "intelligent" (for "brain waves") these can be further modified to represent:

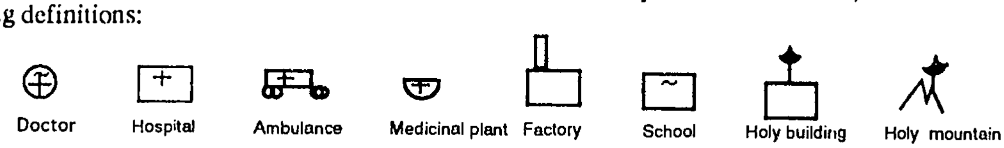


7.2 Applying the Principle of Orthogonality

The term "orthogonality" was introduced by Aad van Wijngaarden in the design of programming languages, to mean features that carry across from one context to another (Wijngaarden, 1965). To ease learning we shall build into MUSLI as much orthogonality as possible.

For example, we can define:

If we now introduce the additional attributes: + (Medical), ♠ (Holy) and ▯ (Manufacturer) together with the symbols to represent a four wheeled vehicle and to represent a mountain, we can now form the following definitions:

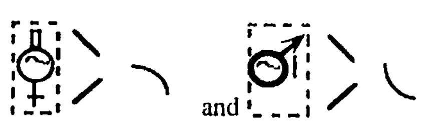


We can introduce additional conventions that will communicate variations in such attributes as age (an increasingly thick outline) or sickness (a cracked or broken symbol).

7.3 Condensing a Group of Symbols

If the attributes in a group are relatively constant then the original symbol plus attributes can be replaced by a condensed short-form. We shall use a frame, [ ], to group primitives and > to represent condensation.

E.g. a working (manufacturing) intelligent woman condensing to: ⌝ , and a wise old solitary man (i.e., a hermit) condensing to ⌞, is represented by:

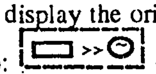


Compound symbols can be condensed at any stage either by the reader pausing and clicking on them or by the system converting them. Usually the reader will need warning before a symbol condenses and this can be achieved in a variety of ways. Perhaps the compound symbol will have its border change colour before the change or both symbols may be displayed at once with the compound one gradually decreasing in size while the condensed symbol increases. Interesting possibilities of letting an electronic personal assistant control the degree and timing of the condensations remain to be explored.

The condensed symbols will be animated just as groups of primitives can be. A working (manufacturing) woman carrying water may be either  or  and might be shown moving towards a growing (manufacturing) plant . We would then, of course, see the attribute for growing increase in size!

Again we emphasise that we are not producing cartoons: the symbols, like those shown above, will have to be learnt. They are not intended to be immediately obvious, but to carry meanings that will not be forgotten once learnt.

7.4 Expanding Condensed Symbols

Expanding symbols is the reverse process to condensing symbols. As mentioned in Section 3, readers will be able to freeze the display at any point in time and then, for example, by double-clicking on a compound symbol display the original primitives. Thus double-clicking on the symbol for a vehicle, , may expand it back to:  – where >> is the symbol for motion (since a vehicle is an inanimate object moving people).

8. Bringing it All Alive With Dynamic Symbols

At this point the reader may be forgiven for wondering whether we are not simply reinventing a system of hieroglyphics. The difference is that people such as the Egyptians did not have ways to animate their written symbols!

In a MUSLI document objects can move and change with time, and hence we frequently refer to it as a movie. Describing dynamic processes is not easy and of course we look forward to the day when we shall be writing a MUSLI movie instead of static text!

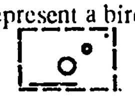
We shall begin by very briefly indicating how we might dynamically represent each of the processes introduced in Section 6. First of all it is easy to see how we can animate the material processes of doing. Symbols can be made to move on the screen in ways that suggest walking, running, jumping, dancing, breaking, and making – for "making" we can show the parts coming together to form a whole. For the cases of giving and taking we suggest introducing a dynamic shaft with an arc at the end of it. On object is "given" by letting the shaft extend and "push" the object away. An object is "taken" when it is drawn in closer to the subject as the shaft shortens.

Secondly, mental processes of sensing, which we represent by dynamic lines, will continuously change in position, length, width, colour and intensity. The golden shaft of love, reaching out towards its object, may become more or less intense in colour as the movie continues. A child's initial fear of a doctor will be palpably obvious when shown by a pulsating black jagged arrow and with time the fear may fade (literally) and morph into the symbol for respect.

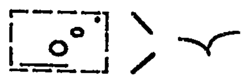
Thirdly there are the relational processes of being. Besides the attributes that are assigned as associated symbols, as described in Section 7, there are the emotions, which we can indicate by shading the object different colours. Here again the dynamic nature of our medium allows emotions to change in a continuous spectrum – as of course they do in real life. Conflicting emotions can be beautifully depicted by showing an animated interplay between the colours representing rivalling emotions – both colours and intensities can interact.

Attributes can also be dynamic. A symbol that depicts an animal slowly ascending a mountain can certainly suggest a very high mountain.

Going one step further, symbols themselves may be formed from animated sequences. A simple circle representing an animal "flying" off a base line can represent a bird flying,



Of course this can be condensed:



And again, the condensed symbol can be animated – in this case by letting the wings move. And so on!!

The modifiers may also change with time. The symbol representing "immature" may morph into the symbol for "sophisticated" as the character in a story gains experience.

At this point we would like to change our point of view to the "other side of the mountain" and make a brief reference to the language for cosmic intercommunication: LINCOS (Freudenthal, 1960). The author hypothesises that the "reader" can deduce meaning from the given graded series of examples. New words are introduced one at a time and the hypothesis is that if the number of examples read is large enough then the reader will deduce from its context the intended meaning for each newly introduced "word" (in this case a word is a pattern of radio pulses). This principle can obviously be applied to MUSLI movies, particularly those that contain symbols for abstract ideas. Although LINCOS deliberately avoids "showing" as a means for illustrating meaning, MUSLI can use dynamic symbols to represent abstract concepts such as number. We can successively display two plants, two animals, two vehicles, and so on, before condensing the dynamic symbol down to a static one. Once the numeric symbols have been introduced we can then dynamically introduce the symbol for the set of natural numbers, etc. The LINCOS exercises introducing the rules of number would in themselves make appropriate MUSLI movies for introducing algebra.

We can also use dynamic symbols for other abstract ideas such as "human". In this case the dynamic symbol would show various subsets of the set [man, woman and child] before condensing to the symbol for human. An idea like "medical" can be formed by demonstrating its adjectives.

We are very aware of how much we have left undone but we look forward to delving deeper. We refer the reader to the paper "DynamIcons as Dynamic Graphic Interfaces: Interpreting the Meaning of a Visual Representation", (Jonassen, Goldmann-Segall, & Maurer, 1993), for an exposition on moving icons.

9. Examples

It is very difficult in a printed paper such as this to describe what are essentially highly dynamic processes. We shall outline just a few.

Our first abstract document, produced in Macromind Director, was an exceedingly simple yet surprisingly convincing movie of a sick child's visit to hospital. The animated symbols show clearly the bonding between the mother and child. The trip into the hospital and the child's recovery were also easily animated.

In our current project, a little more ambitious, we plan to produce a presentation, and possibly a CD version, of a MUSLI movie which depicts a city plagued by smog. The interactions between citizens and administrators lend themselves well to dynamic representations. At the beginning the interactions are overtly hostile but they change dramatically with time. The solution to the crisis is a novel one as it neither bans personalised transport nor enforces public systems[1].

We would like to see a hypermedia system implemented where the user can be provided with interactive MUSLI movies corresponding to alternative search paths through the network – complete with symbols indicating the type of links that are used and what kind of data is available on the way.

And imagine how much more enjoyable it would be coping with e-mail presented in a MUSLI setting with dynamic symbols – particularly if we had an electronic personal assistant to help us!

10. Conclusion

Even a superficial survey of signs and symbols shows the richness of our semiotic heritage. The system we propose will be flexible enough to blend traditional forms of documentation into modern multimedia information retrieval systems while gradually becoming enriched with its own symbolic language. MUSLI

---

[1] The small low-powered individualised units which are used for short trips can be coupled together, as required, with the rightmost vehicle assuming control. For longer journeys the various units, together with their passengers, are driven onto high speed public transportation vehicles. Individual drivers and passengers relax in their own seats or take advantage of on-board facilities such as dining cars and boutiques and may even find time and opportunity for an overdue haircut!

enables "reading" at different levels using an extended version of a reference system, and makes maximal use of the principle of orthogonality to aid learning. We discuss the necessary general requirements for a symbolic language before making a first concrete proposal based on the principles of functional grammar. We are aware that the mastery of any new language involves effort, but surely we can take heart from the success of other interactive activities such as video games.

The system we have described is a first step towards mastering the art of electronic communication involving all our senses, so that we can reap the real benefits associated with a new age of hypermedia.

It seems appropriate to conclude a paper such as this with a quote from a Gary Larson cartoon. The medieval soldiers are fighting to grim death on the battlements of the castle. Arrows and spears are flying thick as rain. Then one soldier turns to another and remarks that it doesn't really matter whether they win or lose: it's fun! This is how we feel about MUSLI: it is not yet clear whether the eventual outcome will be a revolution in con munication and language, or a modest addition to the theory of semiotics. One thing is clear: MUSLI is fun.

We hope that this first glimpse will encourage many readers to join us in further exploration of MUlti Sensory Language Interfaces.

References

Alty, J. L. (1993). Multimedia: We have the technology but do we have a methodology? In H. Maurer (Ed.), *Proceedings of Ed-Media 93 – World Conference on Educational Multimedia and Hypermedia, Orlando, Florida* (pp. 3-10). Charlottesville, VA: AACE.

Cypher, A. (Ed.). (1993). *Watch What I Do: Programming by Demonstration*. Cambridge, MA: MIT Press.

Dreyfuss, H. (1972). *Symbol Sourcebook*. New York: McDraw Hill.

Freudenthal, H. (1960). *LINCOS Design of a Language for Cosmic Intercourse*. Amsterdam: North-Holland.

Halliday, M.A.K. (1985). *An Introduction to Functional Grammar*. Baltimore: Arnold Ltd.

IEEE, (1993). Visualization [Special issue]. *IEEE Computer Graphics and Applications*, 13(4).

Jean, G. (1989). *Langage de signes, l'écriture et son double*. Gallimard.

Jeanes, R. (Ed.). (1982). *Dictionary of Australasian Signs for Communication With the Deaf*. Melbourne: Victorian School for Deaf Children.

Jonassen, D. H., Goldmann-Segall, R., & Maurer, H. (1993). *DynamIcons as dynamic graphic interfaces: Interpreting the meaning of a visual representation*. Institutes for Information Processing, Graz, (Rep. No. 376).

Liungman, C. G. (1991). *Dictionary of Symbols*. Santa Barbara, CA: ABC-CLIO.

Magnenat-Thalmann, N., & Thalmann, D. (1991). Complex models for animating synthetic actors. *IEEE Computer Graphics and Applications*, 11(5), 32-44.

Maurer, H. (1992). *Die andere seite des berges*. Essay 142, Maurers Meinung, Austrian BTX *Maurer ###4.

Maurer, H. (1993a). From multimedia to hypermedia. In J. Hosking (Ed.), *Proceedings of the 13th New Zealand Computer Society Conference* (pp. 2-9). Wellington: NZCS.

Maurer, H. (1993b). *Spekulationen Über die Multimediale Zukunft*. CAP Debis.

Maurer, H., & Carlson, P. (1992). Computer visualization, a missing organ, and a cyber-equivalency. *Collegiate Microcomputer*, 10(2), 110-116.

Maurer, H., Kappe, F., Scherbakov, N., & Srinivasan, P. (1993). Structured browsing of hypermedia databases. In *Proceedings HCI'93, Vienna* (pp51-62). Berlin: Springer. LNCS 733.

Nielsen, J. (1990). *Hypertext and Hypermedia*. Boston: Academic Press.

Schank, R. C. (1975). *Conceptual Information Processing*. Amsterdam: North-Holland.

Schank, R. C., & Abelson, R. P. (1977). *Scripts, Plans, Goals and Understanding*. Hillsdale, NJ: L. Erlbaum Associates.

Umiker-Sebeok, J., & Sebeok, T. (1987). *Monastic Sign Languages*. Berlin: Mouton de Gruyter.

Wijngaarden, A. (1965). Orthogonal design and description of a formal language. *Mathematisch Centrum*, Amsterdam: MR 76.

Zissos, A. Y., & Witten, I. H. (1985). User modelling for a computer coach: A case study. *International Journal of Man-Machine Studies*, 23, 729-750.

A Study of Button Theory in Structuring Human-Computer Interaction in a Multimedia System

CHEE-KIT LOOI
Information Technology Institute, 71 Science Park Drive, Singapore 0511

JOON-KEONG CHAY, LOUISE CHEW & KANE CHAN
Department of Information Systems and Computer Science, National University of Singapore Kent Ridge Crescent, Singapore 0511

Abstract: Much research has been directed towards the interactivity of computer-aided instruction (CAI) software. *Button Theory* allows the student to express his feelings and questions to interact with the computer at the touch of a button. thus enhancing control over his learning process in a CAI environment. We describe the initial results of our study in implementing *Button Theory* in a directive learning environment and offer some comments on its genericity in different learning environments from a designer's perspective.

Introduction

In most computer-aided instruction (CAI) software as well as other computer-based learning environments today, students generally lack control over the instruction they receive. When students feel that their learning needs are not being met, learning becomes passive for them, often resulting in boredom, frustration or a dislike for learning through computers.

Button Theory advocated by Roger Schank's group at the Institute for the Learning Sciences [Jona et al, 1991], provides the student using computer-based learning environments with as much control over what they see, hear and learn. *Button Theory* is implemented by means of a comprehensive set of messages, which the student can use to interact with and control the computer-based tutor. Rather than using natural language processing (which is difficult), each message corresponds to one button, represented iconically on the computer screen.

This paper reports on a project to explore the feasibility of incorporating *Button Theory* in a hypermedia CAI system. We seek to address this question: how feasible and adaptable is the use of *Button Theory* across different computer-based learning environments. We are currently approaching this from a designer's perspective and have not gone on to doing this from the user's perspective. We have implemented our interpretation of *Button Theory* in a prototype CAI system on astronomy called *Solaria*. Following a detailed description of the interpretation and implementation of *Button Theory* in *Solaria*, we will discuss the genericity of the theory, i.e, we will discuss whether each message (and hence, the button) proposed in the theory can be general enough to be used in different domains. The *Solaria* prototype is programmed in HyperTalk 2.0, the scripting language for the Macintosh HyperCard 2.0.

Button Theory proposes three categories of possible discourse between the student and the system. The three categories and the messages in each category are:
Feelings - "Awesome!", "Boring", "No Way", "Huh?", "Too Hard";
Questions - "How Do I Do That?", "Now What?", "What's the Point?", "Why?", "History";
Control - "Change Task", "Back up", "Big Picture", "More Detail", "Skip This."
A full description of the meaning of each can be found in the original paper [Jona et al, 1991]. We will next discuss our interpretation of the *Button Theory* in *Solaria*.

Interpretation of the Button Theory

Out of the 15 buttons proposed by *Button Theory*, 13 buttons which were most appropriate to the context of the domain i.e. astronomy were chosen for implementation in *Solaria*. The main function of these buttons was to act as user responses and control to interact with *Solaria*.

Implementing them, however, was tedious as each of these buttons could be interpreted in many ways. This was due to their flexibility and relative generality. To reduce overlapping interpretations between buttons and to maintain consistency in system response, the interpretations for each button were pared down to only the essential ones. We expect that this streamlining will in no way reduce the responsiveness of *Solaria* nor the interaction between the user and *Solaria*. In fact, having an essential set of interpretations helps to train the user to more accurately express his feelings by selecting the most appropriate buttons. This will help to prevent abuse of or over-reliance on general buttons like "Huh?" and "Why?".

Feelings Group

Awesome The user expresses his enthusiasm for the current topic or task being learned. This button serves basically as feedback to the teacher from the student as regards to the subject/topic under study at that particular point in time. This button which conveys the user's enjoyment provides an important opportunity for the teacher to gather useful information about the student's preferences and interests.

Too Hard The user is suggesting that the material presented is too hard for comprehension. As such, he will like to be given the option to restart the same topic again at a lower level of complexity. At that level, the difficulty level of the material will be less and the style of presentation will also be simplified.

Boring The user is suggesting that the material presented is too easy and that he is getting uninterested. As such, he will like to be given the option to restart the same topic again at a higher level of complexity or presented with more interesting material on the current topic. At that level, the scope of the material will be wider and the topic will be studied in greater detail.

Huh? "Huh?" is a very general button that can be easily abused. Therefore, the interpretations were limited to these few essential ones which are, as far as possible, not duplicative of other button interpretations. In fact, special care has been taken to ensure that its interpretations do not clash with those of the other general buttons, such as "Why?" and "What's the Point?"

i) Under normal circumstances, the user is indicating a complete lack of understanding to the material being presented and would like a further or alternative explanation.

ii) If this button is selected after some animation or display, it is similar to i) and the user is seeking further clarification on the animation or display. It should be noted that for some of the queries the user is asking, the more appropriate button to select should be the "Why?" button. The "Huh?" button acts as a catch-all safety net which will handle the rest of the queries which are not covered by "Why?".

iii) If this button is selected right after some recommendation has been made to perform some action, the user is indicating that he does not know how to go about performing the recommended action. For example, Solaria can suggest to the user to click on an icon of the Halley's Comet to see an animation. At this point, if the user clicks on "Huh?", it indicates that he does not know that he is supposed to move the cursor and select the comet icon by shifting the mouse and clicking on the button.

Questions Group

History The user needs to know all the topics that he has covered so far. He is presented with a sequential list of these topics covered.

How (do I do that)? After viewing certain animation or experiment being conducted, the user will like to given a step-by-step demonstration on how to duplicate the experiment.

Now What? The user is expressing that he is unsure on where to proceed to or what to try out next and will like to be provided with some recommendations.

What's the Point? Depending on the circumstances, there are two separate system responses provided. Even though the responses are different, selecting this button can still essentially be viewed as asking the question: What is the point of that?

i) If selected right after a recommendation was made for the user to perform or try something, it is indicative that the user wants to know how this suggested action relates to the topic being taught.

ii) If selected after some animation or display, it is indicative that he needs to know how the presented relates to the topic being taught. Basically, the question being asked is: What is the point of having me do that? For instance, under the topic of gravitation, an animation was shown of the planets moving around the sun. By clicking on "What's the Point?", the user is asking: "What is the point of having this animation shown within the topic of gravitation?" The system response would be: "To illustrate the forces of gravitation at work in our solar system."

Why? This is one of the more general buttons and can have one of these interpretations:

i) Under normal circumstances, the user is implying that he does not understand or that he does not agree to the logic of the material being presented and would like further clarification.

ii) If this button is selected right after some animation or display was presented, it is similar to (i) and the user is seeking further clarification on the animation or display. For instance, under the same example given under "What's the Point?", an animation of the planets moving around the sun was shown. By clicking on "Why?", the user might be asking: "Why do the planets move in uniform ellipses around the sun?" The system response would be: "This is due to the gravitational forces between the sun and the planets."

iii) If this button is selected right after some recommendation has been made to perform some action, the user is looking for a relation between the suggested action and the topic under study. He is basically asking the question: "Why should I do that?" This is in fact similar to "What's the Point?" under the same conditions.

Control Group

BackUp The user feels a need to review the previous material covered before continuing.

Skip The user is finding the current topic uninteresting, difficult or boring, and wishes to proceed to the next topic. It must be noted that this button does not duplicate the functions of the "Boring" and "Too Hard" buttons. Those buttons returns the user back to the beginning of the current topic at a different level of difficulty.

More Detail This button indicates the student's desire for more information on the current topic.

Big Picture The user feels that he needs to know how the current topic relates to the parent topic or even astronomy.

Genericity of the Button Theory

We will discuss whether each message (and hence, the button) proposed in the theory can be general enough to be used in other domains. By generic, we mean that the code associated with the message or button is operative or easily adaptable in any environment without any major modifications. i.e., it is possible to develop a "template" of codes which can be implemented in any domain. In HyperCard™ terms, there are basically two containers: CONTEXT and RESPONSE. CONTEXT contains the list of conditions in the form of predicates whereas RESPONSE contains a list of statements on how the system should respond when the conditions evaluate to true. The code for generic buttons is composed of pseudo-code statements such as:

If CONTEXT then RESPONSE

The conditions employed by *Button Theory* have been classified into the three main groups of student goals, student knowledge, and communication history, and are expressed as predicates [Jona et al, 1991]. To formulate appropriate or meaningful responses to the messages expressed by the system, the developer of the system needs to analyse all the possible conditions, and then determine suitable combination of conditions which describes all possible contexts in the system. Next, he needs to design suitable responses for each context.

We illustrate by examining the response to "Why?" Let's consider the possible context for "Why?" For example, a student may be presented with a fact (e.g., "the earth revolves around the sun", or "animals fight over territory and food") or asked to do some task (e.g., "drag the pendulum away from the centre") or he may have observed some actions (e.g., a chimp walking up to another chimp and threatening it). The student then asks why with regards to any of the scenario above. When a fact is presented, the condition "Fact-Presented" would be true. When user is asked to do a task, the condition "Action-Asked" is set to true. Similarly for the last scenario. Hence, to determine the context, each relevant condition in the set of conditions is tested. Once the context is determined, appropriate responses can be made.

Types of Learning Environments

As there are many different types of environments of teaching and learning, it would be appropriate to categorise all environments. We look initially at two broad types of learning environments. The first one is directive teaching/learning in which the environments cover instructional or factual materials. An example of an instructional environment would be "How to operate a sophisticated machine" where teaching involves providing the learner with a set of step-by-step instructions to operate the machine. An example of a factual environment generally would be academic in nature such as physics or chemistry or as implemented, astronomy. In this environment type, the facts are presented and taught to the student.

The second category is one of exploratory or discovery teaching/learning. An example would be the *ChimpWorld* environment described in [Jona et al, 1991] to teach the behaviour of chimpanzees. The messages/ responses from the student will differ from the first category. [Chay et al, 1994] provides a comparative analysis of the buttons used in *Solaria* and *ChimpWorld*.

We will next discuss the genericity of the theory in relation to these two types of learning environments.

(a) Some buttons proposed in the theory but are not necessary or relevant

In the implementation of the theory in *Solaria* (one of directive teaching and learning type of environment), 13 of the 15 messages were used. The two messages in the initial theory not used are "No Way" and "Change Ttask". "No Way" was excluded for the simple reason that it is used to express the student's disbelief or surprise of some thing, probably fact, presented. To respond to this expression will be very difficult as the cause of the feelings would be unknown to the system. Hence, we would rather the student express this feeling in another form, probably by asking questions "Why?", etc.

On the other hand, "Change Task" was excluded as we felt that the message was more appropriate for the intuitive or exploratory type of learning where a student can choose to change the task that he is currently doing. In the case for astronomy, the student can easily change the topic of teaching by going back to the menu to select another topic. Thus, messages that deal specifically with intuitive or discovery learning may not be relevant in an environment which is directive or factual in nature.

(b) Some buttons not proposed in the theory but are necessary

There were also some messages which were needed but not proposed. The messages in this category are mainly those of the fact-seeking or fact-clarification type. An example would be when a student is learning about solar eclipses, he would like to know about other types of eclipses (if any) in which case he would need to express the "What are the related topics" message. We tried to circumvent such messages using those that were proposed. Although certain messages can be circumvented or rephrased easily, there were some messages that were quite ambiguous as a result of circumvention. There are yet others that just could not be rephrased in any other way. In the latter case, we propose new messages and hence buttons.

An example of a message that is not supported by the theory is "What makes the Sun appear brighter than the other stars?" or "How is it that the Sun appears very bright?". Both these messages can be circumvented using "Why does the Sun appear brighter than the other stars?". This is an easy example. Others may not be easy.

Two new buttons were used in the system. They were "Glossary" and "Related Topics". Both these buttons are related to fact-seeking as mentioned earlier. "Glossary" provides a simple explanation of a term while "Related Topics" provides curious students with more facts.

One very prominent question or message that is not in *Button Theory* is "What?" It seems that in a directive environment, a significant number of questions can be based on "What?". We also suggest that the "How do I do that?" message be expanded to just "How?". This is to include other "How?" questions such as "How does the earth revolve around the Sun?" or "How is our Sun like other stars?" We however did not experiment further with this suggestion although the need for such a message did arise a few times.

(c) Environment specific buttons

To summarise, there are buttons which are mainly used in a directive environment for learning, and there are others that seem only relevant to an exploratory or intuitive environment. Examples of the former are "Related topics", "Glossary", etc. while examples of the latter environment would be "No Way" and "Change Task". Some other buttons in the theory, which have the same semantic sense in both the types of environments, could very well be different functionally. The "Too Hard" message in a directive environment would mean that the student finds the material too difficult to understand and an appropriate response by the system would be to rephrase the concept/fact in simpler terms. However, in a exploratory environment, "Too Hard" could be used to express the same feeling but the system would respond either by reviewing the material in a slower manner or by asking the student to carry out some active task which would increase his understanding.

Another environment specific button that needs to be mentioned is the "Huh?" button. This button is used as a catch-all. with the student using it to the exclusion of the other buttons. As a result, it is so general and very vague and therefore, by that inherent characteristic, is very environment specific. In fact, this button is not just environment (directive or exploratory) specific but is even domain-knowledge specific. For example, an appropriate response to "Huh?" in physics might be very much different from that in botany, even though in both domains, it conveys the student's confusion.

Other buttons that might be environment specific (and may be to a certain extent domain of knowledge as well) are "Awesome!" and "Boring". As both these buttons are used to express some feelings, it is up to the designer or developer to interpret those feelings and to respond to them in the most appropriate way deemed by him/her. For example, we decided at some times to interpret "Boring" to mean that the material was too easy while at other times, the material was too dry. The response to the former was to present further materials with a higher level of difficulty while the response to the latter was to either tell a story, myth or legend relating to the material presented or simply ask the user to skip the topic or skip to a harder level (see diagrams in next page). In summary, it seems that all the buttons under the "Feelings" category of messages are environment specific buttons. This is quite logical considering the nature of the category.

(d) Environment independent buttons

Basically, the buttons not in the previous category belong to this category. This would mean that all the buttons in the "Questions" category and all but one button (i.e., "Change Task") in the "Control" category are environment independent. This may not be surprising for buttons in the "Control" category.

However, for buttons in the "Questions" category, one would expect that since the messages in the category pertain to the domain, this would inherently imply that such buttons are also environment specific. However, we concluded otherwise. The main reason for this is that in order to make appropriate or meaningful responses to the messages expressed by the student, the system needs only consider the contexts that the student is in when such messages are expressed. Indeed, the contextual elements proposed in *Button Theory* (e.g. Current-Goal-Explained or Current-Goal-Not-Explained, Action-Not-Explained, Repeated-Last-Button-Press) contain no predicates that relate directly to the domain [Jona et al, 1991]. In any given environment or domain, the context can be derived from such a set of determinable conditions. All possible conditions must be thoroughly analysed by the developer. Appropriate responses to messages can then be made by examining the relevant conditions.

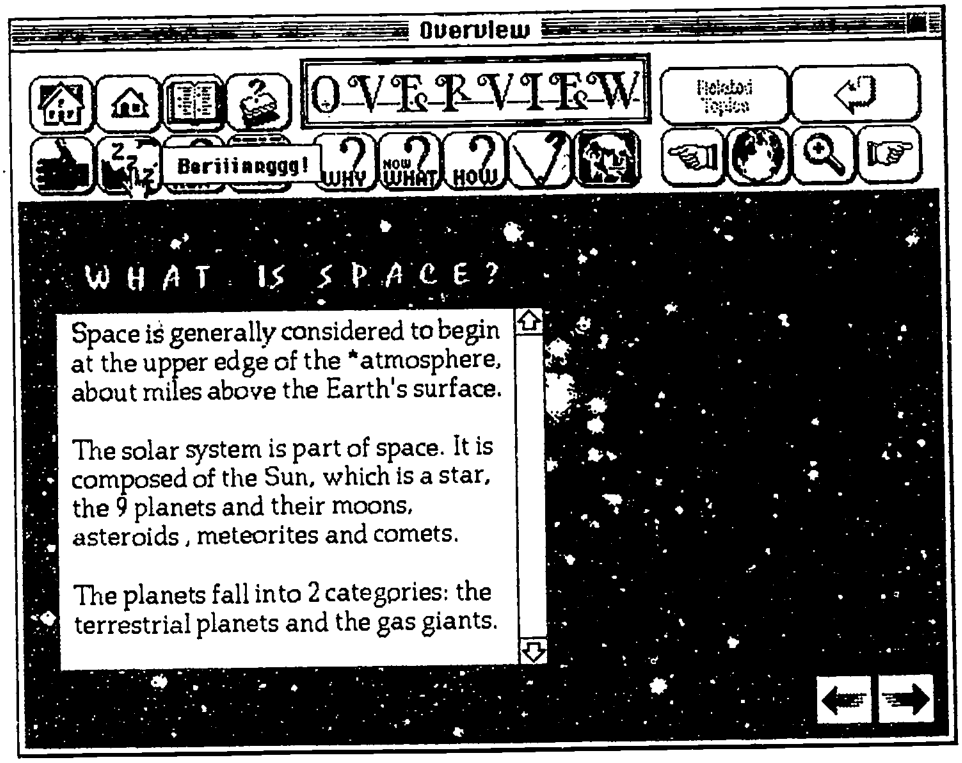
Conclusion

In summary, this study suggests that the buttons/messages in the Control and Questions categories can be generic while those in the Feelings category are less generic. One implication of this is that the development of a learning environment based on *Button Theory* would be simplified. We need to acquire a core set of conditions or contextual variables on which to select a response to a button press [Jona et al, 1991]. Templates of codes for the responses need only to be developed once. For a different domain, these codes can be copied and incorporated into the system. The developers of the system need only provide additional contextual variables, if needed, and the rules by which the buttons (and the system) respond to the user. A definite area for further work will be to conduct studies of actual students using the system. This will provide further empirical studies on *Button Theory*. Another area to investigate is the extent to which *Button Theory* ease or increase the cognitive load on the students that is created by "navigating" while trying to learn, compared with other types of user-interfaces.

References

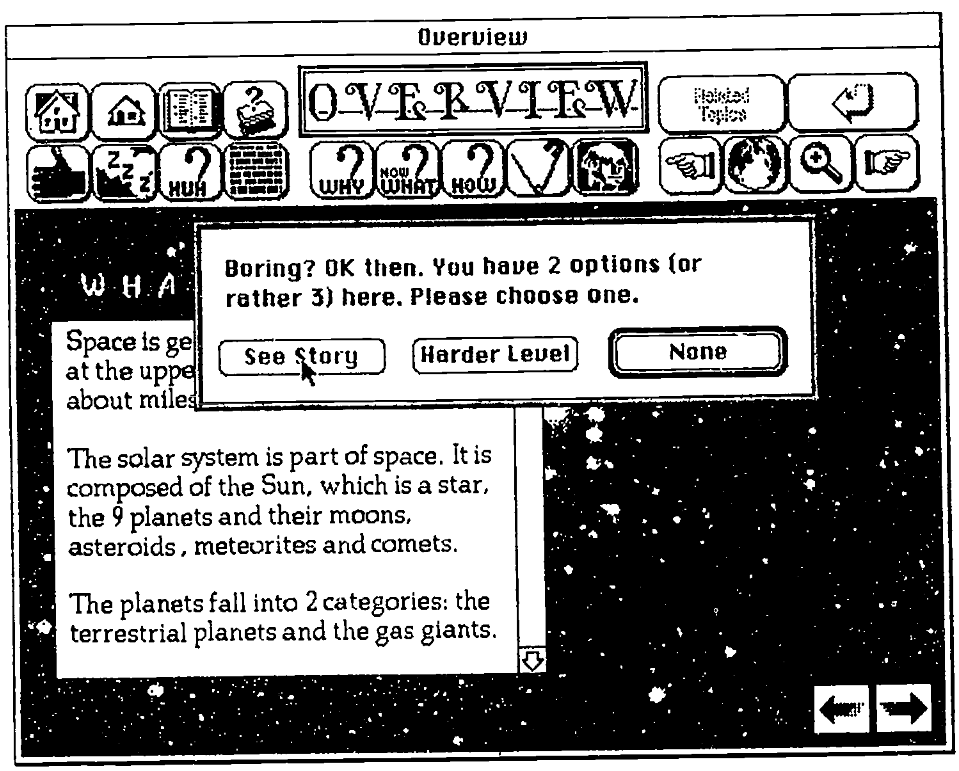
Chay, J.K., K. Chan & L. Chew (1994). A Study of the Genericity of *Button Theory* in Different Learning Environments, internal report, Department of Information Systems & Computer Science, NUS, Jan 1994.

Jona, Menachem, B. Bell & Birnbaum (1991). *Button Theory*: A Taxonomic Framework for Student-Teacher Interaction in Computer-Based Learning Environments, Northwestern University, The Institute for the Learning Sciences, Technical Report #12, Evanston, Illinois.



Environment-dependent card 1

To demonstrate the use of "Boring" button
User clicks on "Boring" button.



Environment-dependent card 2

He is presented with a choice of actions. He selects the "See Story" option.

A Knowledge-based Learning and Testing System for Medical Education

Siobhan MacDonald
School of Business
Carleton University, Ottawa, Ontario, K1S 5B6, Canada
E-mail: smacdona@business.carleton.ca

This study involves research on the development and the testing of a prototype of a patient simulator. Development of the prototype required three main stages: analysis, design and implementation. This paper will focus on the design and implementation stages of the study which were recently completed. The present version of the simulator runs on a Macintosh Quadra 800 and has a window-based interface with a limited use of graphics. The field-testing of the simulator on medical students showed us that the prototype would be useful in testing a student's diagnostic and treatment skills, however it would be advantageous to enhance the interface with audio and visual effects.

I. Introduction

The current medical school curriculum has been questioned in numerous studies, the main areas of concern being course instruction and student evaluation (Barrows, 1983; Anbar, 1987; Barker, 1988; Warner, 1989). New methods of teaching and testing the skills of medical students must be developed to ensure that students have the necessary medical knowledge and clinical skills to meet current societal health needs. The traditional medical curriculum and internships must be supplemented by standardized teaching modalities such as computer-assisted instruction (Carbonell, 1989; Fattu and Patrick, 1990).

This paper focuses on the design and implementation stages in the development and testing of a patient simulation system. This system was designed based on the restructurable modelling method and implemented using Negoplan, a knowledge-based shell (Szpakowicz & Kersten, 1993). The system's reasoning mechanisms are described in Kersten *et al.* (1993). MacDonald (1993) presents the development methodology, the knowledge bases and the results of field testing.

II. Knowledge-based Simulation

Simulation is a method of analyzing a problem situation by creating a model of the situation that can then be manipulated by trial-and-error methods (Fishwick, 1991). Patient simulators can be defined as a representation of a clinical situation in which an individual conducts the diagnosis and management of a patient (Assman, 1979).

Patient simulators possess many advantages: they allow students to have access to conditions which may not be routinely encountered in the clinical setting, they pose no risk to an actual patient, they provide an enhanced sense of reality, they provide immediate feedback, and they are capable of evaluating a student's response.

Current knowledge-based patient simulators possess some but not all of the above-mentioned advantages (Patient Simulator II, 1989; Parker & Miller, 1989; Champetier DeRibes, 1989; Johnson, 1989). The presented patient simulator differs from other approaches in that it possesses all of the advantages mentioned above. The patient simulator is presented through a discussion of the design and implementation stages of the study.

III. Design

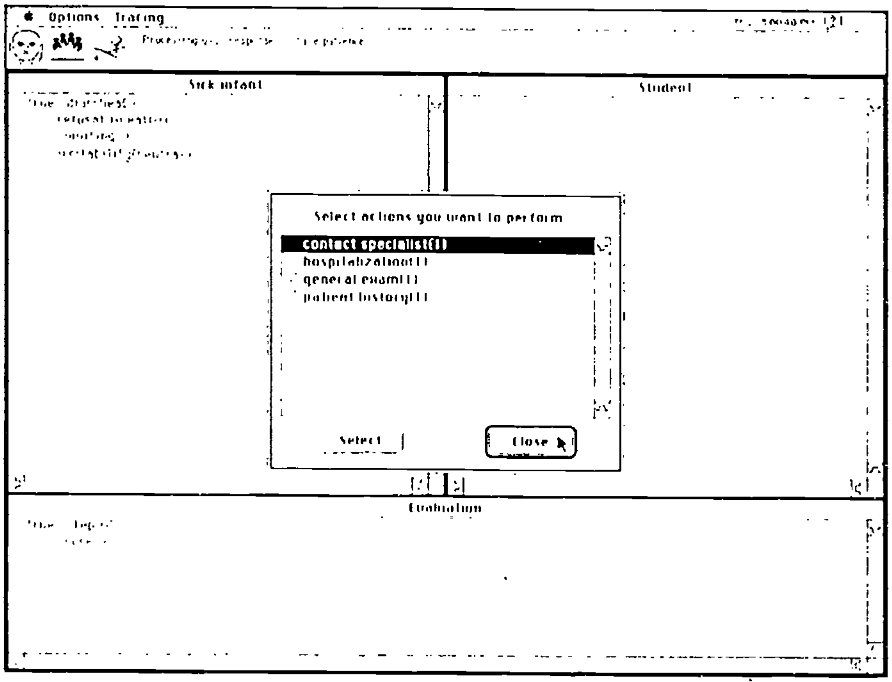
The design stage commences once the analysis stage has been completed. The main purpose of the analysis stage was to determine the user's requirements and to elicit and model knowledge. The cases modelled in preparation for encoding were: a viral intestinal disease, a bacterial intestinal disease and a bacterial neurological disease. These cases were selected because the visible symptoms are similar for each disease, therefore making the diagnostic process more challenging for the students.

The main purpose of the design stage was to encode the models of knowledge. The shell chosen to implement the patient simulator was Negoplan (Szpakowicz and Kersten, 1993). Negoplan is a knowledge-based shell based upon the principles of restructurable modelling, therefore allowing for changes in the case management process by merging these changes with the case management model, resulting in a new representation of the patient. The patient simulator can be run with a scoring system if so desired. The scoring system is designed to allocate points for each step in case management. The student following the shortest route to diagnosis would receive a perfect score.

Figures 1-5 show screen snapshots from the patient simulator running a typical case management scenario. The scenario demonstrates the shortest path to the diagnosis of a bacterial intestinal infection. The first window (Sick Infant) displays the patient's condition and displays the results of the actions selected by the student. The second window (Student) displays the actions that the student selects. The third window (Evaluation) displays the scoring system and the step in the case management process.

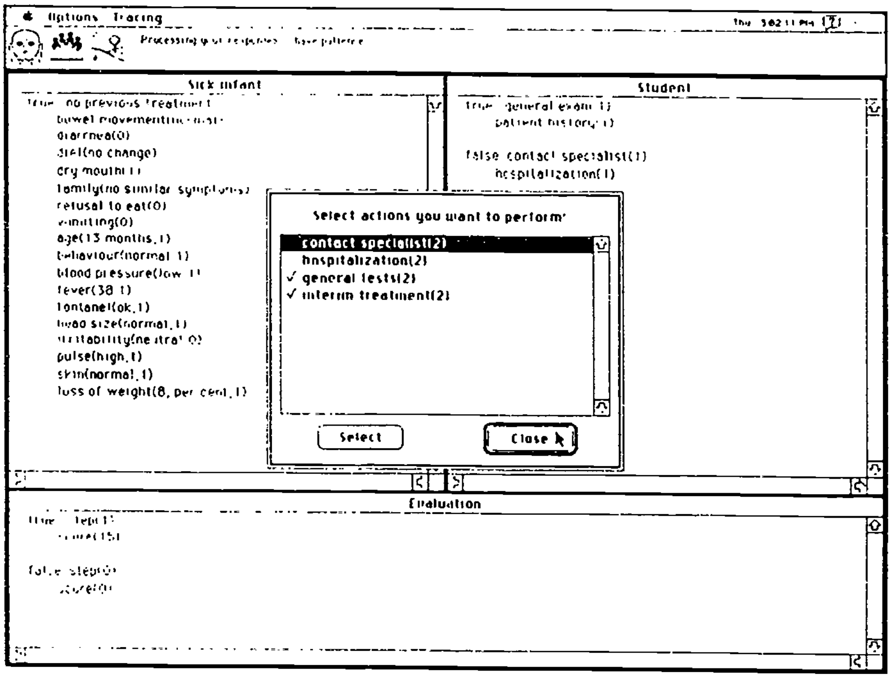
In Fig. 1 the student is presented with the initial representation of the patient, and based on the symptoms, makes the first selection to inquire about the patient's history and to perform a general exam as is indicated by the checkmarks.

Figure 1. Initial Representation and Selection 1



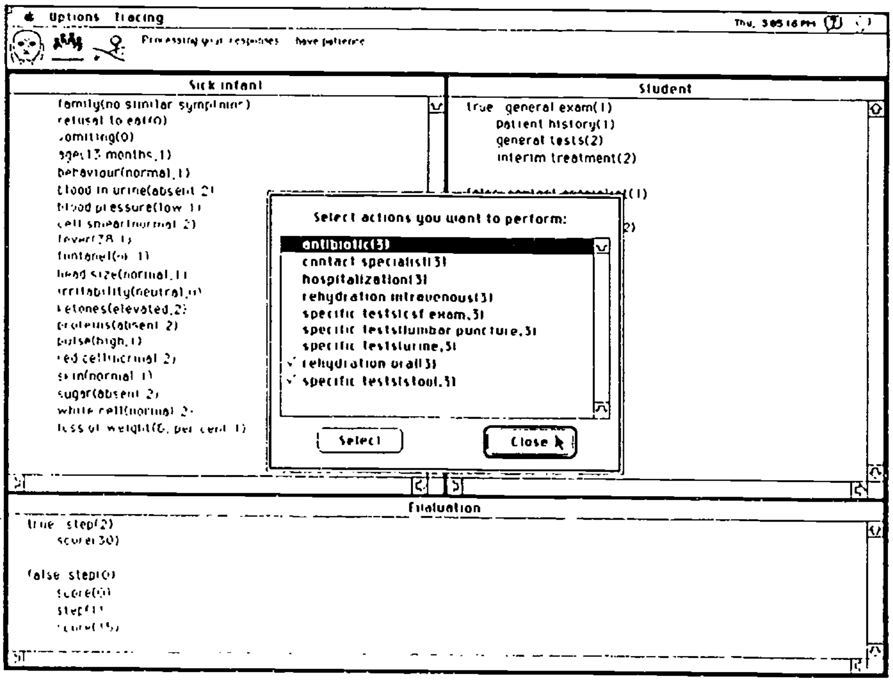
The results from the student having selected the patient's history and a general exam are presented in Fig. 2. The student selects general tests to aid in the diagnostic process and selects the provision of an interim treatment to alleviate the patient's symptoms. As is indicated in the evaluation window, the score is currently fifteen.

Figure 2. Modification 1 and Selection 2



In Fig. 3 the representation of the patient changes once again and the results of the general tests are displayed. The test results are normal, with the exception of the level of ketones which is elevated, indicating a possible infection of the intestinal tract. The student chooses oral rehydration as the interim treatment and a stool test as the specific test. The score is now thirty.

Figure 3. Modification 2 and Selection 3



After the third modification, the results of the stool test are displayed as can be observed in Fig. 4. In addition, the interim treatment has alleviated the infant's symptoms. The diarrhea and the vomiting have disappeared. At this point the student has sufficient information to make a diagnosis of a virus of the intestinal tract. The score is forty-five.

Figure 4 Modification 3 and Selection 4

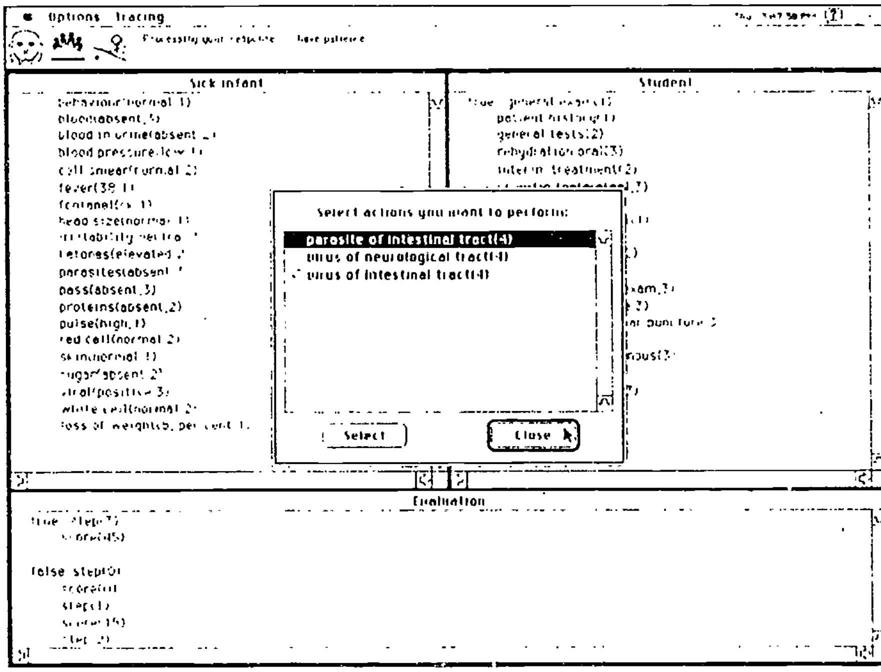
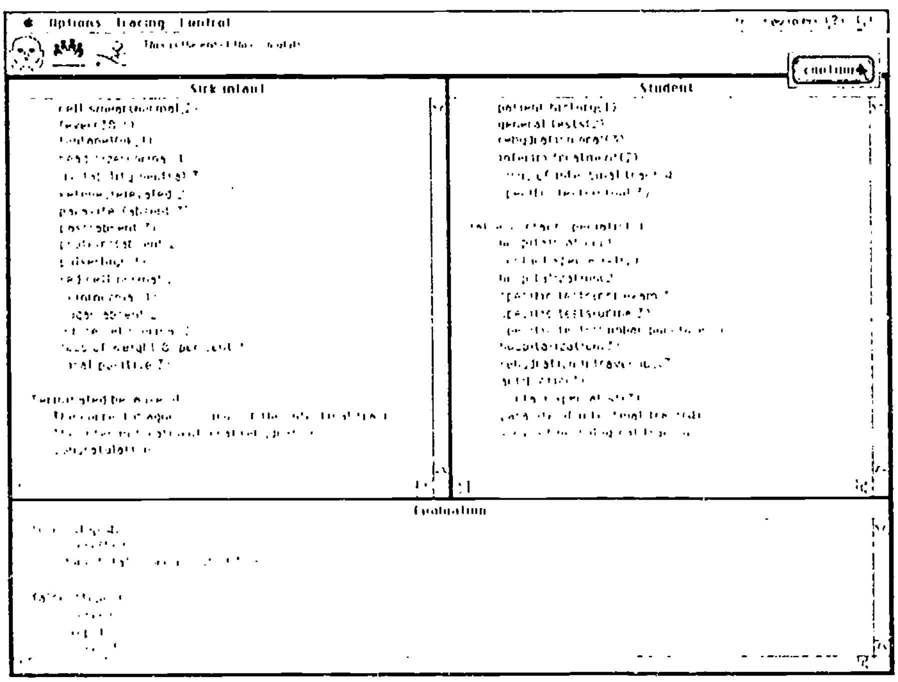


Fig. 5 depicts termination of the simulation in which the student is informed that he/she has made the correct diagnosis and has provided the correct interim treatment therefore receiving a perfect score of fifty.

Figure 5. Modification 4 and Termination



IV. Implementation

The field evaluation of the patient simulator was performed on an ongoing basis. Twelve students from varying academic fields evaluated the patient simulator based on the following factors: system ease of use, system consistency and speed, diagnosis and treatment scenario representation, accuracy of the clinical problem simulation, accuracy and value of the feedback and the effectiveness of the student evaluation modules.

The main conclusion drawn from the testing sessions was that the patient simulator is a useful tool in testing the diagnostic and treatment skills of medical students. However the majority of students commented that the patient simulator could be improved by enhancing the user interface. The user interface, although functional, requires further development to ensure its full acceptance in the field of medical education.

V. Conclusions and Future Work

The knowledge bases are still being expanded in order to provide as diverse a testing environment for medical students as possible. The knowledge base is being further developed to include additional infantile infections, specifically, endocrine and metabolic diseases, neurological and psychiatric diseases, and cardiovascular diseases. Current infections in the knowledge base are being expanded to include more detail on the patient's symptoms, the patient's history, the general exam and tests, and the specific tests.

The current user interface will be further enhanced through the following forms of multimedia. First, graphical images will be added to the patient simulator through the use of a scanner that will provide the student with a visual display of specific components of the case management process. Second, actions will be selected through icons which will provide the student with less structured responses than does a predetermined list as exists now. This scenario more closely resembles the case management process. Third, results of the exams and tests will be provided in a sorted order and in a form similar to how a physician receives results in a realistic medical setting, for example, receiving lab results in numerical format thereby allowing the student to make his or her own deductions. Last, sound output will be used to illustrate the behaviour of certain organs such as the infant's heartbeat.

The rationale behind expanding the current user interface design is twofold. Most importantly, the testing scenario with which the student is presented should mirror a realistic medical scenario to provide the student with as rigorous a testing process as possible. In addition, in order to maintain the student's interest in the patient simulator, it is important to make the user interface both useful and competent to ensure the student's confidence in the system.

As the patient simulator is further developed through the enhancement of the user interface and the expansion of the knowledge base, testing will continue and will grow more extensive as the system nears completion. Although much research has been conducted in the area of computer-aided learning programs, there remains a gap in the study of knowledge-based simulation in medical education. With the further development and testing of knowledge-based patient simulators, these tools could provide solutions to many of the current problems facing medical educators today.

References

Anbar, M. (1987). The Socratic Teacher - Computer-Assisted Self Instruction. *Computers in Medicine*. Computer Science Press, 203-225.

Assman, D.C. *et al.*. (1979). *Clinical Simulations for Respiratory Care Practitioners*. New York: Springer-Verlag.

Barker, P., and H.Yeates. (1985). *Introducing Computer-Assisted Learning*. New York: Prentice/Hall.

Barrows, H. (1983). Problem-Based, Self-Directed Learning. *JAMA*, 250, 22, 3077-3079.

Carbonell, J.R. (1989). AI in CAI: An Artificial Intelligence Approach to Computer-Assisted Instruction. *I.E.E.E. Transaction on Man Machine Systems*, 190-202.

Champetier De Ribes, D. *et al.*. (1989). SIMUL: Training for Medical Diagnosis. *International Symposium of Medical Informatics and Education*, 277-280.

Fattu, J.M. and E.A. Patrick. (1990). Computer-Aided Diagnosis and Decision-Making. *Clinical Applications of Computers in Medical Care*, 201-233.

Fishwick, P. and R. Modjeski (eds.). (1991). *Knowledge-Based Simulation*. New York: Springer-Verlag.

Johnson, L. (1989). The Dental Diagnosis and Treatment Project. *International Symposium of Medical Informatics and Education*, 367-370.

Kersten, G.E., S. MacDonald, S. Rubin and S. Szpakowicz. (1993). Knowledge-based Simulation for Medical Education. *Modelling and Simulation: Proceedings of IASTED International Conference*, M.H. Hamza (ed.), Anaheim, CA: IASTED, 630-633.

Kersten, G.E., W. Michalowski, S. Szpakowicz and Z. Koperczak. (1991). Restructurable Representations of Negotiation. *Management Science*, 37, 10, 1269-1290.

MacDonald, S. (1993). *The Development and Testing of a Knowledge-based Patient Simulator*. Master's Thesis, Ottawa: Carleton University, 1993.

Parker, R. C., and Randolph Miller. (1989). Creation of Realistic Appearing Simulated Patient Cases Using the Internist-1/Q.M.R. Knowledge Base and Interrelationship Properties of Manifestations. *International Symposium of Medical Informatics and Education*, 281-286.

Patient Simulator II. (1989). Computer Software, Knowledge House, 1989.

Szpakowicz, S. and G.E. Kersten. (1993). Recent Developments in Negoplan. *Proceedings of the Annual International ,TIMS/ORSA Conference*, May 1993: 45-48.

Warner, H.R. (1989). Iliad - Moving Medical Decision-Making Into New Frontiers. *International Symposium of Medical Informatics and Education*, May 1989: 267-270.

Acknowledgments

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

Evaluating COCA - What do teachers think?

Nigel MAJOR
Department of Psychology, University Park, Nottingham, NG7 2RD, UK
Email: nigel@psyc.nott.ac.uk

Abstract: This paper discusses an evaluation of COCA, a system that gives teachers control over domain material, teaching strategy and meta-teaching strategy. The purpose of this evaluation is to study more fully the effectiveness of the system. Ten subjects were given an authoring task. The resulting knowledge bases, together with a questionnaire, made up the experimental results. This study shows the strengths and weaknesses of the COCA approach and whether it has helped improve teachers' attitudes towards AI. The results show that the system was successful, yet too complex. Results are tentative due to the size of the experiment.

1 Introduction

A lot of work has investigated the efficacy of different teaching styles in the classroom. Examples include the Plowden Report (1967), looking at teaching methods used in primary schools, studies of formal and informal teaching styles, such as Bennett (1976), and studies of the teaching styles of computer-based training (Eaton & Olson, 1986). This work suggests that many teachers have rejected educational software as it uses the wrong teaching style and is not available in the right subject area. Only recently have intelligent tutoring systems (ITSs) really been able to address these problems, as systems have begun to emerge which offer a range of teaching styles (Spensley, Elsom-Cook, Byerley, Brooks, Federici, & Scaroni, 1990) with tools allowing the teacher to change those styles. Consequently, there is little or no actual evaluation of ITSs in the classroom which support the teacher with a range of teaching strategies, and few or no empirical studies have been published (Murray, 1993). The issue of evaluation has been highlighted as needing attention for ITS systems in general (Brna, Ohlsson & Pain, 1993), with an increasing need for researchers to use appropriate methods for evaluating their results.

COCA, which consists of both authoring tools and a runtime shell, is a system intended to provide teachers with genuine access to ITS technology. Teachers are either uninformed about using AI techniques in schools, or do not have the resources to use them, or more seriously, do not trust the decision making of a system which would be controlling some of the teaching in their classroom. As the result of a number of interviews and lesson observations, Major (1993a) reported requirements from teachers about what they need from an intelligent assistant. Predominant amongst these was the ability to control the teaching style of the system. As a result of these requirements, and also on the basis of the use of meta-level reasoning in the fields of planning and knowledge-based systems research, a prototype system COCA-0, was built. This was then informally evaluated with teachers, resulting in a more complete system, COCA-1, described in Major & Reichgelt (1992), which gives a great deal of control over the teaching style of the system to the teachers themselves. The initial evaluation of COCA was the reconstruction of the teaching strategies of other systems described in Major (1993b). Further, in Major & Reichgelt (1991) a teacher used COCA-1 to build a simple ITS for algebraic equations. This case study demonstrated that a teacher can develop a simple ITS with COCA's authoring tools.

This paper describes a more formal evaluation of COCA-1, comparing school teachers and university teachers, and contrasting their attitudes towards using AI in the classroom before and after the experiment. The subjects in the experiment were asked to perform a given authoring task and to fill in a questionnaire. Both the knowledge bases produced during the task, and the completed questionnaires, were analysed against the aims of the experiment. The purpose of the experiment was to evaluate a more detailed use of COCA's authoring tools, and to discover if attitudes towards using AI in the classroom had been improved as a result of using COCA.

The university teachers were familiar with ITSs although none of the subjects was experienced in using COCA. Some of the school teachers had used computers before and some had no computer experience.

2 Background

KAFITS (Murray & Woolf, 1992), is a set of authoring tools enabling teachers to build ITSs directly. The tools allow domain knowledge and tutoring strategy knowledge to be created and edited. KAFITS facilitates teacher involvement in the ITS design process, alongside the knowledge engineer. COCA, whose aim has been to allow the teacher to work alone with the system, is necessarily a less complex system. KAFITS has been evaluated in a case study with one high school teacher and two education graduates (Murray, 1993). This was very much a qualitative evaluation but nonetheless gave some encouraging results in terms of productivity, with about 85 hours of effort needed for an hour of instruction.

Although there have not been any other evaluations of ITS construction tools, a certain amount of work has been done on suitable evaluation for ITSs, which naturally concentrates more on the use of the system by students rather than teachers. Different evaluation techniques have been studied (Mark & Greer, 1993) in order to point towards evaluation methodologies for ITSs. This work is relevant to COCA as it discusses the evaluation of different aspects of an ITS architecture, and most notably, the teaching strategy. It suggests changing the teaching strategy while keeping other components fixed as a possible means of evaluating effectiveness. Of all the techniques discussed, formal experimental techniques are put forward as suitable for summative evaluation of different components of the system.

Educational impact (Littman & Soloway, 1988) is a major test of an ITS in terms of both achievement and effect, expressed by how well a student learns and what attitudes are retained after that learning. Although we are not evaluating the ITSs built with COCA, we can still use these criteria with teachers to see whether they can use COCA's tools successfully, and also what effect this has on their attitude towards using AI tools in the classroom. These ideas are reflected in the aims of this evaluation of COCA.

2.1 General evaluation methods

A general discussion of different methods for evaluating software usability is given in Macleod (1992). One of the quickest methods of evaluation would be to give the system to teachers, asking them to use it as much as possible, and to write a report. This helps find important problems with the system, but is very dependent on the teacher. It is also similar to the case study reported in Major (1993a) and so would be of less value to us here.

Analytic evaluation methods are used early in the design process. They may employ a model of the user and apply it to a specification of the system. This model basically predicts the different interactions likely to be performed by the user. The advantage is very cost effective evaluation, done before implementation.

Controlled experimental studies of software usability are difficult because there are so many dependent variables when it comes to using a computer, such as the users themselves and their attitudes, the tasks they perform and the different environments in which they work. The standard procedure is to separate groups of subjects and to get them to perform the same experiment with one crucial variable changed. The results of the different groups are used to prove or disprove hypotheses about that variable. But isolating one such feature in COCA's context would be a very difficult exercise. An assessment of the value of controlled experiments with regard to evaluating software is given in Monk (1985).

Survey methods (Chin, Diehl & Norman, 1988) are the cheapest form of obtaining evaluation information. They are usually done by questionnaire. Quantitative data can be obtained if more than ten subjects are used, as well as qualitative data. Such a method would certainly be applicable with COCA, as long as the questions are directed towards the aims of the study.

A final set of methods are known as observational methods. This involves watching users working with the system, either directly or by the use of video, and recording the process. This information can then be analysed. This is more costly than using a questionnaire, but allows better access to the task itself. One such technique incorporates a think-aloud protocol (Wright, Monk & Carey, 1991).

The methods most applicable to COCA are survey and observational methods. The survey can be used by asking direct questions about COCA's usability and sufficiency, and about attitudes towards rules for representing teaching strategies. COCA allows users to save their knowledge bases and these can be analysed to allow observational evaluation. Further users could record their interaction with COCA by noting all the decisions they make and the functions they use when performing a task.

3 Experimental aims

The primary aim of the experiment was to assess whether or not COCA will be useful to teachers and authors. A number of issues were considered which are numbered below:

1 A first issue was the teacher's attitude towards AI techniques as opposed to traditional computer-assisted learning. The experiment investigated the teachers' perceptions of the changes in their attitude. It should be noted that this is a subjective measure. Although it could be suggested that the teachers' perception of their attitude change will always increase with an environment change, we observed in Major (1993a) that a system not offering control over teaching decisions would have been a serious disincentive to teachers with respect to using AI in the classroom.

2 Another question of particular interest was whether teachers considered they had had a conscious view of their strategies before using COCA. This is also a subjective measure.

3 Further, the experiment set out to discover whether teachers felt that the heuristics provided by COCA were likely to be sufficient to control teaching behaviour.

4 At a more practical level, the experiment set out to see which of COCA's facilities in the opinion of the teachers, were easy to use, and which were little used and difficult. Further, it attempted to assess which areas of authoring support subjects felt were likely to have to be extended in future versions. This is important when considering how future research can best build on COCA.

4 Evaluation method

The material chosen for the task was the domain of the American Revolution, which together with its associated strategy and meta-strategy, formed part of a conference demonstration of COCA (Major and Reichgelt, 1992). This teaching material, which is largely declarative in nature, makes use of a broad range of the features of COCA, including user-defined attributes, non-computer based tasks, hints and meta-strategic rules, and was constructed with the help of a history teacher who did not carry out the experiment. The fact that the subjects did not have any historical expertise is not relevant as we are testing COCA-1's authoring tools and not the final ITS produced.

There were 10 subjects consisting of 5 university teachers and 5 school teachers. Everyone completed the task and the questionnaire, but only 4 school teachers and 3 university teachers returned knowledge bases. The school teachers were all new to COCA and had not been used to find the requirements in Major (1993a). They were less experienced with computers than the university teachers. The subjects had received no training with COCA except for 3 school teachers, who had had 2 hours training together.

The subjects were presented with the domain material, initial teaching strategy and meta-strategy, in the form of prepared knowledge bases, together with a set of instructions in English for altering the teaching behaviour of the final system and extending the domain material. These changes involved creating rules, editing existing rules and using the meta-strategy. As they carried out the task, subjects were asked to write down each step that they took, including mistakes, using some examples on the instruction sheet to guide them as to what level of detail was required. The purpose of this interaction record was to give an indication of the complexity of the task. It would give some idea of how long the task took and how many mistakes were made. The task had been performed by the author, thus giving a measure of solution efficiency by comparing the number of rules to achieve given results.

Once the task had been performed, the subjects were asked to return their saved knowledge bases for analysis. They were also asked to fill in a questionnaire about the general use of COCA, the suitability of the teaching strategy representation, the domain representation, and a number of questions about how the subjects' opinions of AI in general and COCA in particular had been changed as a result of the task. As many questions as possible were given a closed set of possible responses, with 100mm scales being widely used. The aim of the analysis was to determine whether or not COCA was useful to teachers and authors, how attitudes towards AI were affected by using COCA, how conscious teachers were of their own use of strategies and whether or not COCA's heuristics were sufficient for teachers. All of these points were specifically addressed in the questionnaire.

5 Results

5.1 Knowledge bases produced

The knowledge bases produced were tested and analysed to see how much of the task had been completed successfully and to assess solution efficiency, by comparing with an optimal solution. Table 1 shows the relative success in the task.

The mean success rate for the whole task was 68%, which shows that the subjects were reasonably successful but nonetheless had some difficulty using COCA-1's authoring tools. The most difficult task was moving the summary to appear at the end of the teaching about a particular object rather than at the beginning. This involved deleting a rule from a group of rules that the subjects otherwise did not use. Consequently the success with this task was poor. Also difficult was the use of user-defined attributes. This coupled with feedback from the questionnaire suggests that the interface to COCA's domain authoring tools was not satisfactory and shows how to improve future versions of COCA.

The first four subjects were all school teachers. Subject 2 in particular seems to have needed considerable training with the system before being able to make any reasonable progress with it. Those subjects with the least computer experience did least well, which is an expected result. The time taken by each subject varied between 2 and 3 hours. Notes taken by the subjects while doing the experiment showed that there were few repeated attempts to perform a particular task, but rather once an initial attempt had been made for a task, it was usually abandoned.

Table 2 gives us some idea of the relative efficiency of the solutions produced by the subjects. This is achieved by comparing subjects' solutions with an optimal solution which solves the tasks making the minimum number of changes. It should be noted that the optimal number of changes is only optimal in terms of efficient editing, and makes no claim as to its efficiency as a teaching solution. The number of changes to each knowledge base was calculated by counting one change for each addition, edit and deletion. The score shown in the table is the number of correctly completed tasks, as shown in table 1.

KB	Task Description	Subjects							$\bar{x}$	σ_{n-1}
		1	2	3	4	5	6	7		
Domain	Change name	Y	Y	Y	Y	Y	Y	Y		
	Add objects	Y	Y	Y	Y	Y	Y	Y		
	Place objects	Y	Y	Y	Y	Y	Y	Y		
	User attributes	N	N	N	Y	Y	Y	N		
Subtotal (%)		75	75	75	100	100	100	75	86	13
Strategy	Change name	Y	Y	Y	Y	Y	Y	Y		
	Stay after fail	Y	N	Y	Y	Y	Y	N		
	Fill in blank test	Y	N	Y	Y	Y	Y	Y		
	Categorise test	Y	Y	Y	Y	Y	Y	Y		
	Test on dates	Y	N	Y	Y	Y	N	Y		
	Move summary	N	N	N	N	N	N	N		
	Swap date/causes	Y	N	Y	N	Y	N	N		
	Teach user attributes	N	N	N	N	N	N	N		
Subtotal (%)		75	25	75	63	75	50	50	59	19
Meta-strategy	Change name	Y	Y	Y	Y	Y	Y	Y		
	Remove implications	Y	N	N	Y	Y	Y	N		
	Restore implications	Y	N	N	Y	N	Y	N		
Subtotal (%)		100	33	33	100	67	100	33	67	34
Total (%)		80	40	67	80	80	73	53	68	16
Score (max 15)		12	6	10	12	12	11	8	10.1	2.3

Table 1: Percentages of tasks achieved by each subject

The efficiency for each subject is then calculated as the number of completed tasks per change to the knowledge base made, relative to this number for an optimal solution.

	Subjects							
	Optimal	1	2	3	4	5	6	7
Domain changes	4	3	3	3	4	4	4	3
Strategy changes	13	7	5	10	9	11	8	6
Meta-strategy changes	3	3	1	1	3	3	3	1
Total changes	20	13	9	14	16	18	15	10
Score	15	12	6	10	12	12	11	8
Efficiency $\frac{(\frac{Score}{Changes})}{(\frac{Score_{opt}}{Changes_{opt}})} \times 100\%$	100	123	89	95	100	89	98	107

Table 2: Relative efficiency for each subject

We can see that although we have only an approximation to the efficiency of the solutions, that they are all very close to the optimal solution. The reason that it is possible to be more efficient than the optimal solution is that the more difficult tasks, which are completed in the optimal solution, require more changes to the knowledge bases than the majority of the tasks performed by the subjects. So a subject who has not attempted the more difficult tasks will have made fewer changes in proportion to his score than the optimal solution, and thus will have a higher efficiency. Subject 2 scored relatively highly despite not achieving many of the tasks. This was because the tasks she did achieve were done efficiently, and she did not make many attempts at those she could not manage. A lower efficiency also shows that a subject has taken longer to perform the task in comparison to what was achieved. Generally, we can see that the subjects managed to make the changes to the knowledge bases without using more rules than necessary, and so that COCA-1's flexibility is not at a high cost in terms of long editing sessions.

5.2 Questionnaire findings

Some of the questions asked for a mark on a 100mm scale, whereas others asked for more subjective opinions about teaching strategies and the use of AI in general. Table 3 gives the results. The scores are given with 0 as the worst possible score and 100 as the best. The average and standard deviation are also given. The subject numbers correspond to the numbers in tables 1 and 2 and are included for reference.

The first 5 subjects are school teachers and the rest university teachers. Firstly, and most generally, we can see that the majority of averages are over 50, putting them in the positive half of the scale as far as COCA is concerned. Question 3.3, regarding the use of variables in rules, shows the largest standard deviation. This results from the university teachers suggesting that they were strongly in favour of variables, whereas school teachers felt they would be highly unlikely to use them. On the other hand, question 3.8, regarding the practicality of building libraries of teaching strategies with COCA for general use, had the lowest standard deviation and the highest degree of consensus. This shows the potential of a system like COCA within a school or other teaching environment. Other points to notice are the high

Qu.	Description	Subjects										$\bar{x}$	σ_{n-1}
			1	2	3	4	5	6		7			
1.2	Terminology clear	35	50	27	44	74	88	85	94	75	72	64	24
1.3	Confident usage	41	40	6	40	31	61	86	22	82	55	46	25
1.4	Getting lost	57	17	75	44	64	68	75	61	52	46	56	17
1.5	Broad usage		87	50	30	51	89	85	95	23	73	65	27
1.6	UI easy	34	51	36	37	70	21	83	93	66	90	58	26
1.7	Tools useful	79	51	31	54	95	90	83	94	97	86	76	23
2.3	Domain tree clear	80	38	56	43	96	100	99	96	15	94	72	31
2.7	Course libraries	-	-	92	69	89	-	78	32	93	96	78	23
2.8	Domain distortion			-	29	87	-		67	96	17	59	35
3.3	Likely to use variables	64	51	3	58	4	98	100	93	-	83	62	37
3.6	Suits your teaching	70	81	95	66	89	30	18	44	93	-	68	23
3.7	Decisions easily made	24	80	-	63	92	11	87	26	92	86	62	33
3.8	Strategy libraries	-	89	99	81	93	93	98	86	94	83	91	6

Table 3: Answers to questions using a 100mm scale

scores, particularly amongst the school teachers, to question 3.6, regarding the completeness of COCA's teaching strategy model with respect to their own teaching style. The only score to be strongly negative was that of question 1.3, regarding whether users often felt lost in the system. It is clear that COCA's user interface needs to be improved to increase users' confidence and stop them getting lost.

The next part of the questionnaire measured changes of attitude towards teaching strategies and AI in the classroom. We shall concentrate on our school teachers as it is their attitudes that are perhaps more important. The two questions of interest to us are question 1.8, regarding the use of AI as opposed to CBT in the classroom, and question 3.10, which asked about attitudes towards teaching strategies. Also relevant are some of the comments made in the unstructured comments section of the questionnaire. Without exception, all our school teachers felt that COCA had shown them that AI could be useful in the classroom and had something to offer. They of course mentioned the problems of the interface and terminology, but could see the underlying usefulness of the system. With regard to teaching strategies, none of the school teachers had had as structured a view of strategies as COCA. Four of the five felt that forcing the user to consider their teaching in such a way was useful. None suggested that this view of teaching made authoring either difficult or distorted.

A number of other points were made in the responses to the other more open questions. The main concerns with the system were its terminology, documentation, user interface, student model and domain representation. The first three of these would be the main effort in any development of COCA's authoring tools for commercial use. Although the student model is weak it could easily be extended to allow any teacher-defined attribute (psychological/pedagogical) to be given to a student and to control the teaching. The domain representation could also be extended, with perhaps the strategy interpreter giving the domain interpreter more control, thus allowing larger and more interesting pieces of domain material to be taught. Some other strengths of COCA that subjects mentioned were its potential for formalising teaching through the strategy model, and thus to become a useful tool for trainee teachers. Subjects thought that being forced to consider teaching decisions beforehand would be profitable, and would help very much with the mixed-ability problem in the classroom. Finally they felt that the ability to change the student rating thresholds (i.e. the student modelling behaviour) was very useful.

6 Concluding discussion

It is certainly the case that COCA-1 is a complex system, yet despite this complexity, the experiment has shown that the ideas and structure contained in COCA-1's authoring tools are worth carrying forward, although the tools themselves need to be made easier to use. The experiment has also provided insight into the attitudes of teachers towards using AI, and in particular the use of teaching strategies.

Although the number of subjects was not large enough to make any categorical statements, it does give us a basis upon which to examine the different attitudes towards COCA, and also to examine differences between subjects. Those subjects who were university teachers were typically much more experienced in the use of computers and were familiar with AI techniques. Despite the breadth of tasks the majority of subjects completed well over half the tasks, with school teachers not doing particularly worse than university teachers. The high standard deviation on the meta-strategy tasks does suggest that such a level of abstraction above the domain was difficult for some subjects. Further the efficiency of use was very good by all subjects. This is an encouraging point especially for naive users.

In relation to the first experimental aim, COCA improved the perception of their attitude towards AI amongst the school teachers from neutral to positive. We can conclude then that the everyday use of tools like COCA for allowing AI models of teaching to be manipulated by teachers is likely to be a fruitful means of improving the acceptance of AI in schools. The university teachers opinion of their change in attitude was that it was not changed by the experiment in that they remained in favour of using AI.

The second aim of the experiment concerned the subjects' opinions of their views on teaching strategies. Those subjects who did not feel that they had had an explicit concept of a strategy were typically the school teachers. They suggested that the formalising of a process that had previously been implicit was useful. The university teachers felt they had had a conscious view of strategies before.

With regard to the third aim, that of the sufficiency of COCA's strategy heuristics, the subjects said that COCA was indeed sufficient for their requirements. Further, there was strong support that building up libraries of strategies would be a practical way of using COCA. Those subjects who already had an idea of using strategies, typically university teachers, felt that COCA's model of a strategy and facilities for building strategies were sufficiently flexible for their needs. Indeed no subjects suggested aspects of the teaching process that were not catered for in COCA's tools.

The final aim was to discover those aspects of the system that were easy or difficult to use, and thus which aspects might need to be extended. With regard to this aim the experiment has also shown a number of weaknesses with COCA. We have already mentioned the domain representation and student modelling. Other points included the need for better and more comprehensive documentation and improvements in the interface. None of the subjects felt the teaching strategy model was a weakness. However, the fact that a number of tasks were not completed suggests that COCA's authoring tools need to be made more intuitive.

In summary we can say that COCA is usable enough for simple tutoring systems to be built, suggesting that a system like COCA should be pursued further, particularly with regard to the points arising from the final aim of the experiment. The school teachers who were suggesting the use of strategy and domain knowledge base libraries were highlighting a real problem for a system of COCA's type, namely that power is required at the strategy level, and yet the authoring task must be very simple if teachers are going to use a system on an everyday basis. To address this, a new version of COCA is under development, which very much simplifies the authoring task by hiding all rules from the teacher and giving graphical controls for strategy construction. This new version, running under Windows 3.1 on a PC, will be used as the basis of further experiments to investigate the nature of meta-strategic knowledge.

References

Bennett, N. (1976). *Teaching Styles and Pupil Progress.* London: Open Books.

Brna, P., Ohlsson, S. & Pain, H. (1993). Preface. In *AIED-93*, University of Edinburgh. AACE.

Chin, J.P., Diehl, V.A. & Norman, K.L. (1988). Development of an Instrument Measuring User Satisfaction of the Human-computer Interface. In *CHI-88*, 213-218. ACM.

Eaton, S. & Olson, J. (1986). "Doing computers?" - The micro in the elementary curriculum. *Journal of Curriculum Studies*, 18, 343–344.

Littman, D. & Soloway, E. (1988). Evaluating ITSs: The cognitive science perspective. In Polson, M. & Richardson J. (Eds.) *Foundations of intelligent tutoring systems.* Hillsdale, NJ: LEA.

Macleod, M. (1992). *Usability Now!* An Introduction to Usability Evaluation. National Physical Laboratory.

Major, N.P. & Reichgelt, H. (1991). Using COCA to build an intelligent tutoring system for simple algebra. *Intelligent Tutoring Media*, 2(3/4), 159–170.

Major, N.P. & Reichgelt, H. (1992). COCA - a shell for intelligent tutoring systems. In Frasson, C., Gauthier, G. & McCalla, G.I. (Eds.) *ITS 92. Lecture Notes in Computer Science, 608.* Berlin: Springer-Verlag.

Major, N.P. (1993). Teachers and Intelligent Tutoring Systems. In *The Seventh International PEG Conference*, Heriot-Watt University, Edinburgh.

Major, N.P. (1993). Reconstructing Teaching Strategies with COCA. In *AIED-93*, University of Edinburgh. AACE.

Mark, M.A. & Greer, J.E. (1993). Evaluation Methodologies for Intelligent Tutoring Systems. *Journal of Artificial Intelligence and Education*, 4(2/3), 129–153.

Monk, A. (1985). How and when to collect behavioural data. In Monk, A. *Fundamentals of Human-Computer Interaction.* London: Academic Press.

Murray, T. & Woolf, B. (1992). Tools for Teacher Participation in ITS Design. In Frasson, C., Gauthier, G. & McCalla, G.I. (Eds.) *ITS 92. Lecture Notes in Computer Science, 608.* Berlin: Springer-Verlag.

Murray, T. (1993). Formative Qualitative Evaluation for "Exploratory" ITS Research. *Journal of Artificial Intelligence and Education*, 4(2/3), 179–207.

Plowden Report. (1967). Children and their Primary Schools. *Report of the Central Advisory Council for Education (England).* London: H.M.S.O.

Spensley, F., Elsom-Cook, M., Byerley, P., Brooks, P., Federici, M. & Scaroni, C. (1990). Using multiple teaching strategies in an ITS. In C. Frasson and G. Gauthier (Eds.) *Intelligent Tutoring Systems: At the crossroads of Artificial Intelligence and Education.* Norwood, NJ: Ablex.

Wright, P.J., Monk, A.F. & Carey, T. (1991). Co-operative Evaluation: The York Manual. Department of Psychology, University of York.

Melding Television, Networking, and Computing for Interactive Remote Instruction: Exploiting Potentials

Kurt Maly, C.M. Overstreet, Hussein Abdel-Wahab and A.K. Gupta
Department of Computer Science
Old Dominion University
Norfolk, Virginia 23529

E-Mail: (maly, cmo, wahab, ajay)@cs.odu.edu

Abstract

Televideo instructional techniques have yet to exploit the advantages made possible by continuing cost reductions in video electronics, computers, and communications. In these technologies lies the potential both to reduce costs and simultaneously improve the quality of instruction. For example, Virginia expects colleges and universities to teach more students with no additional faculty but has not suggested how this can be done. Approaches such as what is described here can make this possible, particularly if used for upper division and graduate courses which may have low enrollments unless offered at several locations. The synergistic combination of these technologies can allow a degree of interactive participation among students and teachers – even when located across the state – which currently is only possible when people are in the same room. With computer-assisted control, multi-way video, audio, and text, instructors and students can participate in a virtual shared classroom which supports traditional instructional modes and encourages modes of instruction not previously feasible. The ability to continue the use of traditional, proven instructional techniques as well as supporting development of new modes of instruction in a way which allows faculty, not experts in computer science, time to become competent in the use of new approaches is crucial to the success of the approach. In this paper we describe a prototype system which will enable us to learn what capabilities are most useful and how to use them.

Background

The combination of emerging digital television and computing/networking technologies can, when properly exploited, provide solutions to many problems confronting higher education (DeLoughry 1993, Douglas 1993 and Mudge & Bergmann 1993). The changing costs in these technologies now allow the development of truly interactive remote instruction in ways previously impossible or too expensive to consider. The combination of these technologies can now provide, at reasonable costs, an effective virtual traditional classroom (i.e., one in which all students and instructors present in the same room) in a form which allows not only the use of traditional instructional methods (e.g., lecture, question/answer, audio-visual, or small group discussions) but also new capabilities (discussed below) largely as instructors or students prefer.

Old Dominion University is fully committed to discovering how to exploit these potentials. We currently enroll more than 2,000 students in telecourses and have awarded more that 200 degrees to students across Virginia since 1987. Our televised courses are received at approximately 100 higher education centers, community colleges, high schools, hospitals, and military installations within the state. These courses support a complete baccalaureate program in engineering technology, a nursing program leading to a BSN, and continuing teacher education. We also offer graduate engineering degree programs at 20 sites in Virginia.

However the technology currently used to support these courses is largely the same that has been used for two decades or more (except for the transition to digital transmission); interaction between students and instructors is limited. It largely fails to incorporate any computing or computer networking technologies. Old Dominion University is currently funded by the state of Virginia to explore the feasibility of expanding our existing remote education program by offering 500 courses per year to a total of 12,000 students in 20 different academic degree programs at 30 specially equipped sites. Each site would have the computing and television facilities necessary to support this vision of interactive remote instruction.

This paper describes our vision of how computers, networking, and television can be combined to produce not only a better, more effective teaching environment but also at a reduced cost compared with traditional classroom teaching. It outlines how such a system can be implemented with current technology and how its delivery can be improved with the coming availability of ATM networks. It also gives a description of the functionality of the current prototype system and what technology is used with deployment for a demonstration project. We discuss advantages and disadvantages of the system and some of the decision which went with the implementations of the prototype. We outline our plans for a demonstration project and future expansion to support on offering of hundreds of courses at dozens locations in Virginia.

The Vision

Our goal is to hide the technologies (computer, networking, and television) which enable interactive remote teaching (IRT) whenever possible so that neither instructors nor students need deal with them. Basic instructional modes should be "natural" to instructors and students alike; the overhead of use (e.g., preparation of classroom materials by instructors and students' use of the system) must be minimal, at least for basic modes. The system must accommodate itself to "technology hostile" instructors and students alike. But it should also allow both students and teachers to exploit its additional capabilities when they choose. An individual's use of advanced features should be allowed to easily evolve as she becomes more familiar with system and sees a personal benefit in their use. We believe that the ability to continue use of traditional, proven instructional techniques as well as supporting development of new modes of instruction in a way which allows faculty time to become competent in the use of new approaches is crucial to the success of the approach.

The key to the success of these technologies is that they allow faculty to be more effective teachers. Effectiveness involves both productivity (how much time faculty must spend in preparing classroom material, evaluating student work, and interacting directly with students) and the quality of the educational experience for the students. Computer-aided instruction has long and dismal history in this respect due less to equipment costs than the intensive labor required both to prepare course material and then keep that material current.

The drop in costs for the necessary supporting hardware makes possible the development of truly interactive facilities for educational use. These are not passive TV courses with a simple audio channel for questions; the facilities should allow students to be fully engaged in presentations. But we will have to learn what capabilities are most useful and how to use them.

The current prototype implementation uses both satellite transmission analogue video signal and Internet and Ethernet LANs to transmit data and low quality video. This mix of analogue and digital signal is necessitated more by current costs than availability of technology. At this point it is not clear what technology we will eventually use to transmit information, it may be optical fiber, wireless transmission, ATM or HIPPI and SONET; however, it is clear that to provide a truly effective environment we need to have all information in digital form. Hence in the long-term we see the classrooms containing workstations connected by a network, in whatever form is appropriate, and all the manipulation, control of discourse, and features described below will be done by application software on computers rather than such devices as analogue video mixers.

Current Prototype System Description

The facilities for the full system will include several specially equipped instructional classrooms for student use and a few broadcasting facilities. The student facilities will be located across the state and on the main campus. Currently, the system is being prototyped on the main campus within three classrooms within one building. The regular TV offering at Old Dominion University sends video/audio to remote sites through satellite but in this prototype we use coax cables because everything is within one building.

Each student facility consists of several multimedia workstations, all networked to each other and the main campus and to Internet which will be used for data transfers among various sites. At this stage we use regular 10 Mb/s Ethernet with multicasting IP and a T1 connection to Internet for data traffic. In the near future we will replace the 10 Mb/s Ethernet with a 30 Mb/s parallel Ethernet using Parallel IP (Maly et al. 1993). As a multimedia workstation, each is a general purpose computer and provides a windowing system, has the ability to display both analog television images in a window (delivered to the site by cable or satellite), can display digital video images in some windows (delivered by the computer network), text and graphics in others, can play audio, and has a mouse and a keyboard. Each workstation also has its own video camera which can transmit a picture of the student seated at the workstation and a stylus which the student can use for input. Each classroom will also have one camera which sends an image of the entire room back to any of the central broadcasting facilities.

The instructor's facility will contain similar multimedia workstations as well as at least two additional operator-controlled cameras, one used to project images of the instructor and the classroom from a distance, the other images of material the instructor has chosen to support her presentation: for example, prepared text or graphics, or notes being written while seated at a desk. The instructor can also use her workstation and computer network to transmit notes, graphic images, or animated output of simulations to all student workstations. The system will allow instructors to share software tools with all students (such as spreadsheets, word processors, graphics tools, or simulations) so that all participants can jointly or individually manipulate the same tools (Abdel-Wahab & Feit 1991 and Abdel-Wahab & Jeffay 1994).

Students' view of the Interacting Remote Teaching System

The student can choose mode A, in which the workstation is used most like a typical TV lecture image: most of screen is used to show the instructor and perhaps the whiteboard on which the instructor is writing. The student can also choose mode B in which most of the image is of the material presented by the instructor.

Typical Interactions:

a) *to ask a question of the instructor:* a student clicks on the "Attention" button: the button turns green from red; when the button starts blinking, the students clicks on the button and asks question (the student's image will now appear on all screens); to terminate the connection the student clicks on the button and it turns red.

b) *to answer a question put by the instructor:* the "Attention" button will turn green and blink (and beep); the student clicks and answers then, clicks to terminate.

c) *to operate a tool on instructor's request:* the student clicks on blinking "Tool", then the student can operate the tool while the instructor and all other students watch: the effect of all actions taken by the student will appear on everyone's screen. Examples of such tools are: text editors, spreadsheets, drawings, mathematica, or slide show.

Window Management:

The student can control the windows displayed on her workstation. If the student wishes, she can take a completely passive role and just watch. Students more familiar with the system will be able to operate the windowing systems as they choose to meet individual needs.

Note Taking:

If the student clicks on "Notes"; the whiteboard will appear; the student can type or write with stylus or mouse; click on notes; image at time of second click will be recorded together with notes, notes will disappear after interval.

Playback:

If the student missed class or wishes to review what was presented in class, the student can put the appropriate CD ROM disk in player. Invoke "Interactive Teaching" with playback option.

Instructor's Use of System:

a) *to ask a question of a particular student:* click on "Attention," a class list will appear; the instructor selects a name by clicking the name in the list, student image will appear. Can click on more names to get discussion going; click on "Attention" to terminate interaction. Optionally click on any in class list then all students' Attention will blink; names in class list will blink (those who want to answer); click on one and all others will be disconnected after 10 seconds.

b) *to answer a student:* click on blinking "Attention;" a list of names will appear for whoever has clicked an Attention button; the instructor clicks on one name, then the system behaves the same as when the instructor asks a student a question.

c) *to delegate control of a tool:* click on "Tool;" the class list appears; the instructor selects a student; if desired, the instructor might orally assign an area in computer whiteboard to a student; click to break.

d) *to invoke a particular tool:* type tool name in console. Output produced by tool (wordprocessor, spreadsheet or whatever) will appear on all workstations.

The instructor can take back the floor at any time by clicking.

Typical Scenarios:

Class Discussion:

One or more students initiate a discussion or attempt to ask a question by clicking the Attention button. The Attention button at the instructor's desk will blink and beep. The instructor clicks on the Attention button and selects from the queue of blinking names a student and clicks on it. The selected student's image will appear on screen and that student's Attention button will blink and student can talk after clicking the Attention button. The instructor selects another student to join by clicking on her name in the class list and her image will also be displayed. Fig. 1 illustrates this.

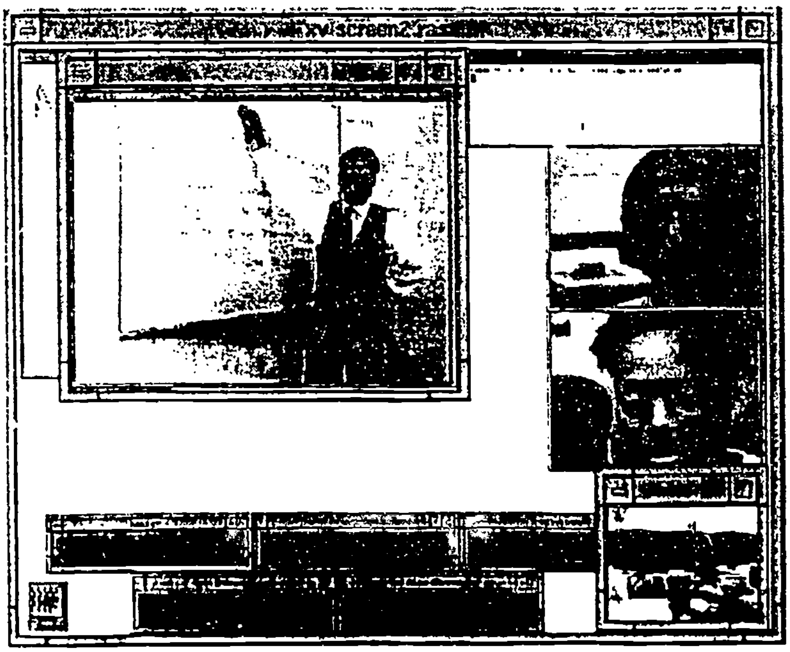


Fig. 1: Workstation Image, Student Discussion

"Joint" editing of English assignment:

The instructor switches to the Computer White Board by clicking on Teaching/Tool and invokes Wordperfect from the Console and places the window in the White Board. She brings up in Wordperfect a file containing an assignment and asks a specific student to show his paper by clicking on Tool and selecting a name from the Class List. That student clicks on his "Tool" button and can now operate that Wordperfect tool and bring up his file for the class to see. The instructor takes back the floor by clicking on her Tool button and makes changes and then dismisses the tool.

Teaching of WorldWideWeb:

The instructor switches to the Computer White Board by clicking on Teaching/Tool and invokes WorldWideWeb by calling the browser xmosaic from the Console and places the window in the White Board. She uses WorldWideWeb to connect to the Library of Congress and searches for publications on virtual reality. She then delegates (clicks on Tool) the tool to a particular student (clicks on Class list) and tells her to search Old Dominion University's Library for books on early childhood learning disabilities. The student takes over the tool and uses it to connect to the library and does the search while all other students see how she performs the task. Meanwhile another student types a note (clicks on Notes) on his private White Board to remind himself to check the availability of xmosaic on his system. The first student gives back the tool (clicks on Tool) and the instructor terminates the session (clicks on Tool).

Discussion

Many states are searching for less expensive ways to meet the demands for both higher education and continuing education, particularly since the enrollment increases which will occur in the next few years will probably come without comparable growth in revenues. Experience has shown that in most cases video tapes of typical college classes are not a sufficient substitute the traditional classroom. But rapidly evolving network technologies will, when coupled with the right tools, enable a degree of interaction among teachers and students which we believe will make the use of distributed classrooms which incorporate both video and networking technologies feasible.

This system should result in minimal loss of the capabilities usually available in a traditional classroom (e.g. question - answer interaction, note taking, group discussion). Instructors who choose to do so should be able to present material in ways similar to what they have done in the past. The instructor is in control in how material should be presented rather than the technology dictating what can be presented and how.

But as important, the system provides the instructor more capabilities such as:

-on-line class lists
-students can see faces of discussants
-integrating computers into teaching
(using tools in class-apprenticeship)
-VCR type playback
-classrooms can be anywhere (but need video and network connections).

While this system has some disadvantages (only talking heads of students, small screens, and if the teacher uses a markerboard, loss of readability can be a problem) and the quality of some video images is largely determined by the capabilities of the supporting network, many of these obvious deficiencies will be addressed when network bandwidth is increased through the use of ATM.

We have used technology to emulate one traditional way of teaching in a distributed classroom environment. We can now use the computers and the networks to truly provide an interactive, collaborative environment for students and instructors and support many modes of instruction, such student presentations, small group discussions. The instructor can provide may kinds of additional information stored in the computer system, e.g., images, text, movies, programs, data, and students can interactively modify this information. These can be used as part of regular classes, support out-of-class assignments, and allow students to review materials individually to support individual learning speeds and needs.

With government and industry interest in networking capabilities, it is likely that the quality of network services, particularly in the form of a significant increase in bandwidth, will continue to improve in the future. The prototype we have described can function effectively with the network capabilities now commonly available, though the use of some types of presentation materials (particularly those relying on digitally transmitted images) will need to be restricted to get acceptable performance. As network services continue to improve however, restrictions on the use of "wide-bandwidth" tools will be relaxed giving presenters more choices in presentations and will support higher fidility images (both in the form of more rapid updates and better resolution).

In addition, once the network exists, it can be used to handle many administrative aspects of the course such as on-line course registration and announcements of various university activities. The instructor can distribute homework, reading lists, video images, and assignments over the network and students can submit their work electronically as well as ask questions of the instructor at any time of the day using e-mail. Since much of the information about the course, the instructor and the students will be available in digital form it will be possible to measure many aspects of the effectiveness of the paradigm. Many of the skills and knowledge students have acquired will be demonstrated on the computer and can be captured electronically for later analysis.

Since our current prototype is limited in its capabilities and has been used primarily to evaluate the effectiveness of existing network services, many key issues are yet to be dealt with. Once more geographically dispersed networks are used, latency may become a problem and require some synchronization between the analog video/audio and computer networked data signals. Until the system has been used by people with other than a computer science background, various aspects of the user interface both for students an instructors cannot be evaluated. These issues will be addressed as our prototype becomes more complete.

Refrences

Abdel-Wahab , H. & Feit, M. (1991). XTV: A Framework for Sharing X Window Clients in Remote Synchronous Collaboration. *Proceedings, IEEE TriComm '91: Communications for Distributed Applications & Systems*, Chapel Hill, North Carolina, 159-167i.

Abdel-Wahab , H. & Jeffay K. (1994). Issues, Problems and Solutions in Sharing X Clients on Multiple Displays. To appear, *Journal of Internetworking Research & Experience*.

DeLoughry , Thomas (1992). Crucial Role Seen for Technology in Meeting Education's challenges. *The Chronicle of Higher Education* (September), 20-22.

Douglas , Suzanne, G. (1993). Digital Soup, The New ABCs of Distance Learning *Educom Review* (July-August), 22-30.

Maly , K., Khanna, S., Foudriat, E.C., Overstreet, C.M., Mukkamala, R., Zubair, M., Yerraballi, R. & Sudheer, D. (1993). Parallel Communications: An Experimental Study. *Second International Conference on Computer Communications and Networks*, San Diego, California, 138-145.

Mudge , J. Craig & Bergmann, Neil W. (1993). Integrating Video and Large-Area Displays for Remote Collaboration. *Technical Report*, Flinders University, Australia.

Directed and Undirected Tasks in Hypermedia: Is Variety the Spice of Learning?

GARY MARCHIONINI and DELIA NEUMAN
College of Library and Information Services
University of Maryland
College Park, MD, 20742 USA
march@umdd.umd.edu mn9@umail.umd.edu

KENNETH MORRELL
Language Center
Rhodes College
Memphis, TN, 38112 USA
morrell@vax.rhodes.edu

Abstract: Questionnaire data from sixteen courses taught over a two year period at eight universities where Perseus was used was analyzed to examine how type of assignment influenced student perceptions about the Perseus interface and learning performance. Assignments were classified as either directed (high instructor control), undirected (high learner control), or mixed and student responses were compared. Assignments that used both instructional types yielded generally better ratings, although directed assignments yielded higher ratings for confidence that needed information was in the corpus and undirected assignments yielded higher ease of use ratings. Additionally, performance ratings were correlated with interface effects but not with frequency of use or previous computer experience.

One of the thorniest problems in applying hypermedia to learning is related to the problem of user control. For more than thirty years (e.g., Mager, 1964) theorists have argued that learners who exercise control over their own learning are more motivated, can more easily relate new information to their personal knowledge, can adjust pacing to personal abilities, and can benefit from learning that more accurately reflects how people learn outside of school. In summarizing the mixed results from many studies of learner control, Ross & Morrison (1989) noted that learner control is a multi-dimensional construct. For example, high ability learners can manipulate instructional as well as presentational aspects of lessons to their advantage, while lower ability learners benefit from presentational but not instructional control.

Hypermedia technology has captured the imagination of educators for several reasons, including its potential for: expanding learner access to information resources, blurring teaching and learning roles, providing a model for associative and non-linear thinking, and offering learner control. In particular, since hypermedia allow users to decide which link to follow, learner control is provided as an inherent feature of the technology. Based on the assumption that learner control is positive and that hypermedia provides such control, many commentators and designers have proposed strategies to leverage learner control in design (e.g., Kinzie & Berdel, 1990;). There is a tension, however, between learning that is fully under the control of the learner (discovery learning) and learning that proceeds according to carefully sequenced and managed activities (training). Duffy and Knuth (1989) pointed out that simply exploring a domain or engaging in "goal free" browsing are not sufficient activities for formal learning settings and argued for careful consideration of what goals hypermedia assignments are to achieve. Most practitioners discover some middle ground, settling either on a guided discovery approach or lessons planned to achieve some specific objective but with the flexibility to take advantage of spontaneous learning opportunities. Interactive technology in general and hypermedia technology in particular have exacerbated the tension between assuring ourselves that students acquire specific knowledge and skills that experts find useful and providing opportunities for students to learn what to learn by making their own learning decisions.

As part of the ongoing evaluation of the Perseus project, questions of learner control were examined through responses of 350 students in 16 sections of courses at eight different institutions on questionnaires specific to their experience with the system. This paper provides evidence for giving both directed and undirected assignments rather than strictly one or the other.

The Perseus Hypermedia Corpus

The Perseus corpus is a HyperCard-based hypermedia information system that contains multi-media information about the ancient Greek world. The system includes Greek and English translations of ancient Greek literature together with various commentaries. The first commercial version (Perseus 1.0) includes the works of Pindar, Aeschylus, Sophocles, Herodotus, Thucydides, Pausanias, and Homer, as well as selections from Plutarch, Hesiod, and Appolodorus. The user can locate particular words in the text and have Perseus provide definitions. Pull-down menus at the top of the screen allow links to other texts, graphics, or various indexes. These are considered implicit links since they require users to initiate hypertext jumps rather than making selections from explicit buttons or anchors that are tied to specific information nodes. In addition to the texts, the project has collected approximately 16,000 images of more than 2000 objects and sites and Perseus 1.0 contains about 7000 of those images. These images represent the collocation of artifacts from a large number and variety of museums and archaeological sites.

A variety of tools make Perseus both usable and useful for analysis. For textual analysis, a morphological parser and dictionary with associated inverted indexes allow users to highlight a Greek word, identify its lemma and find definitions, and quickly see where that word is used in other works. Searches can also be conducted on the definitions themselves to allow users to analyze subtle meanings and styles within and across authors and works. Although a few individual tools exist for scholars to conduct such analyses today, Perseus is unique in providing both aggregation and extension—to the point of redefining how such research is conducted. For archaeological analyses, Perseus offers drawings of sites as they are thought to have existed at different periods of time. These drawings can be superimposed to provide a dynamic representation of developments over time at the sites. Icons on the drawings allow users to display photographs (digitized still images) or short videos (analog videodisc clips) of the sites as they appear today. Users can use the system to create their own interpretations of ideas by designing annotated "slide shows" or multi-media collages (see Crane, 1988; Crane & Mylonas, 1988; Mylonas, 1992 for fuller descriptions of the Perseus system).

Although each of these tools individually represents quantitative changes in what scholars and students can do, there is a synergy in their integration. For example, the literature student can jump to maps or site drawings to clarify and enrich concepts and ideas encountered while reading text. An encyclopedia, an atlas, and bibliographic citations are always available with a few moves of the mouse. Users also have the ability to create and store "paths" through the corpus or to enter notes and annotations. By developing paths, instructors can provide explicit tours through the corpus to illustrate key ideas. These paths can then be assigned to students individually or used in lectures to guide discussion. Students can create their own paths, which are submitted to instructors as interpretations or analyses of assigned problems or themes. A more self-directed use of paths is to facilitate the study of students' own learning by reviewing previous traversals of the corpus in much the same way as learners review passages in paper-based texts that they have highlighted with markers. Perseus empowers users to create new assignments and explorations, new interpretations, and new syntheses; it offers qualitative changes in what scholars and students can do. Assessing the effects of these new capabilities and the new goals they enable requires new and flexible approaches to evaluation and measurement and was a primary challenge of the evaluation effort conducted over the past five years (see Marchionini & Crane, 1994; Morrell, Marchionini & Neuman, 1993; Neuman, 1992 for other results of the evaluation effort).

Method

The methodology used for evaluating Perseus was multifaceted (see Marchionini, 1989, for the "CAT scan" approach to evaluation). In the first several years the main approach was formative (e.g., Flagg, 1990) to provide feedback to the designers. In the later years, more effort was focused on summative results that address how Perseus affects teaching and learning. Data collection methods included interviews, observations, automatic transaction logging, written questionnaires, and document analysis. This paper focuses on results from questionnaires distributed to about 350 students in 16 classes at eight universities over a two year period. These sites represent four large research universities (three public and one private), and four small liberal arts colleges. Table 1 summarizes the different sites, the types of classes in which Perseus was used, and the type of assignment(s) given.

Table 1. Sites, Courses, Questionnaires Returned and Task Type 1991-93

School/Section	Course	Semester	#	Task/Type
1991-2				
1	Clas. world	S 92	33	follow paths (D)
3	Religion	S 92	10	follow path, write essay (M)
4	Literature	F 91	36	conduct lookups, create paths, write essay (U)
5	Religion	S 92	24	follow paths create paths (M)
1992-93				
6	archeology	F92	94	review/enrichment (U)
3	Fresh studies	F92	8	follow path, write essay (M)
5	Religion	S93	5	find info for essay (U)
2	Class/hist	F92	12	follow path, create path (M)
7	Greek lang	F92	3	translations, oral reports (U)
8	Greek lang	S93	9	create paths (U)
1/1	Clas. world	F92	38	follow paths (D)
1/2	Clas. world	F92	10	follow paths (D)
1/3	Clas. world	F92	13	follow paths (D)
1/4	Clas. world	Su92	9	follow paths (D)
1/5	Clas. world	S93	34	follow paths (D)
1/6	Clas. world	S93	8	follow paths (D)

The two-page questionnaire contained both open-ended and closed questions. The closed questions took the form of 5-point Likert scales and were grouped into three clusters: demographic, interface, and performance. All students in the 16 courses were asked to complete a questionnaire at a point in the semester after the Perseus-specific assignment(s) were due. The analysis of interest here was based on comparing questionnaire responses by type of assignment. The assignments instructors had devised were classified into one of three types: directed (D), undirected (U), and mixed (M). Directed assignments asked students to follow paths that were defined by the instructor to find specific information and complete a worksheet or answer short questions. Note that all directed assignments were given at one institution in sections of a course taught by two instructors who each developed distinct assignments. Undirected assignments varied but included using Perseus to assist in translating passages of Greek texts, conducting word analyses in texts to gather evidence for an interpretive essay about ancient Greek values, and creating paths that illustrated themes or concepts. When both types of assignments were given, the assignments were classified as mixed. In the mixed cases, directed path following activities preceded the undirected exploratory activities.

Results

In both years, no statistically reliable differences in students' ratings were found across students in different institutions (large research, large public, small private). Surprisingly, there were no statistically reliable differences due to amounts of previous computer experience or frequency of using Perseus. There were consistently high correlations between students' ratings of the interface (e.g., ease of learning, ease of use, number of times disoriented) and performance ratings (e.g., confidence in finding information, contribution to learning, value of system, satisfaction); but not between demographic ratings (e.g., age, previous computer experience, frequency of Perseus use) and other clusters. Thus, one result of the studies is that the interface is more closely related to students' performance ratings than is previous experience or other demographic variables.

For the questionnaires collected over the 1991-1993 period, one-way analyses of variance across assignment type were performed. Table 2 presents summary data for the ANOVAs. For each of the questionnaire items, results of the ANOVA are shown either as NSRD (no statistically reliable differences) or in the ordered pair(s) of statistically reliable differences at the .05 level based on Tukey HSD post hoc analyses. When multiple pairs of contrasts are listed, they are given in order of smallest to greatest mean effect. For example, in 1992-93 students who did mixed assignments were statistically reliably more satisfied with their learning than those who did undirected assignments, and these differences were smaller than the statistically reliable differences between those who did mixed and directed assignments. The third column gives results for the combined data from both years. Since the 1991-92 questionnaire did not elicit some of the items on the 1992-93 questionnaire, those cells are blank. All effects are reported from most positive to least positive (e.g., for the frequency of getting lost, M>D means that the results were statistically reliably more positive for students in the mixed groups, i.e., they reported being lost less often than those in the directed groups).

Overall, the analyses illustrate that task type was associated with generally strong effects. In general, the mixed task type yielded more positive outcomes and ratings, with the undirected task type showing results that were almost as good in many cases. These results may reflect that students tended to spend more time using Perseus in undirected settings--i.e., when they direct their own learning. Furthermore, these results may be an artifact of the setting since students who did mixed assignments tended to spend more time using Perseus than those in other groups. This possibility is mitigated somewhat by the low of correlation between usage and performance ratings. The results could also be an artifact of the different collegiate settings or may suggest that students generally enjoy more positive experiences when they direct their own learning.

Table 2. ANOVA orderings across task types by questionnaire responses for 1992-3 and 1991-92

Question	1992-93 Responses	1991-92 Responses	1991-93 Combined
Use/Demographic Cluster			
Pfreq (freq of Perseus use)	M>U, M>D	U>M, U>D, M>D	M>D, U>D
Hours (using Perseus)	M>D, M>U		
Interface Cluster			
Learn (ease of learning P)	M>D, U>D	NSRD	M>D, U>D
Use (ease of using P)	U>D, M>D	NSRD	U>D, M>D
Docuse (usefulness of docum.)	NSRD	NSRD	NSRD
Images (quality of images)	M>D	U>M, D>M	NSRD
Lost (frequency)	U>D, M>D	NSRD	M>D, U>D
Recover (ease of recovery)	M>D, U>D	M>D	M>D, U>D
Performance Cluster			
Infoneed (needed info was in P)	D>U	D>U, D>M	D>U
Confid (confidence to find info)	NSRD	NSRD	NSRD
Assndif (difference of assign)	D>U, M>U		
Pcontrib (P contrib. to learning)	NSRD		
Infoamt (amt. of useful info)	M>U		
Value (learning value of time)	M>U, M>D		
Satis (satis. with learning)	M>U, M>D	NSRD	NSRD

Many of the results confirm what might be expected, but other results are more intriguing. Although these results must be interpreted with caution, the following interpretations seem plausible. It seems reasonable that students who worked on mixed tasks or those who worked in exploratory ways spent more time using Perseus than those working on directed tasks. However, it was not the case that students in the directed groups used fewer features than those in the undirected groups (ANOVA for the number of Perseus features used across the three types of tasks yielded an F ratio of 17.04, p=.00, with students in the mixed groups using statistically reliably more features than either of the other two groups and students in the directed group using statistically reliably more features than the students in the undirected groups. Students in the mixed groups used an average of 8.3 features, while those in the directed and undirected groups used averages of 5.7 and 5.0, respectively. Intuition and experience argue that directed tasks are easier to learn and to execute than open-ended tasks. However, Perseus is a complex system and a single diversion from a path can lead users into unknown territory. Students who explore the system may actually expect to become lost, since they are not performing highly goal-directed tasks, and thus may rate ease of learning and use less critically than those who viewed the assignment in a more linear fashion. The differences in assessing ease of learning and use may reflect the fact that students in the directed groups were all from one site, where access to workstations was limited and laboratory assistance was minimal. As is reasonable, the results for frequency of getting lost and ease of recovery parallel those for learning and ease of use. Results for quality of images are mixed: all students rated the images quite highly, and there is little variability across or within groups.

Results for the performance questions illustrate how expectations and task experience combine during learning. Students in the directed groups rated the likelihood that the information they needed was in Perseus reliably higher than the other two groups, since they followed defined paths and accomplished tasks that required them to locate specific information that they knew the designers of the assignments had verified would be there. Students completing more open-ended assignments, however, had to define their own information needs and then locate relevant information by using Perseus finding aids. These students not only worked without knowing in advance whether the appropriate information would be there, they were also subject to the many ways information seeking can go awry during the definition-of-needs step as well as when using such Perseus tools as indexes and word searches. While it is somewhat surprising that directed students did not rate the amount of useful information more highly, this rating may reflect students' lack of awareness of how to locate information beyond that to which they had been directed.

Ratings for how different Perseus assignments were from usual assignments may suggest that students who worked on undirected tasks viewed them as similar to paper-based tasks in which they explored a variety of books or articles for information on topics. Directed students may have viewed the Perseus assignment as more typically linear and pedantic.

Finally, and most importantly, those students who used Perseus in a variety of ways rated the value of the time they spent with Perseus more highly than students in either of the other two task groups. Likewise, they were more satisfied with their learning than those in the other groups. Discovering the proper mix of tasks is clearly one of the challenges of teaching, and these results seem to reinforce a "variety is the spice of learning" view of instruction.

Discussion

As with most issues of pedagogy, the problem of how much learner control is best for students using hypermedia requires relativistic rather than absolute solutions. Although people are adaptable and will learn under most conditions, high levels of learner control yield the best results when learners have good prerequisite knowledge upon which to build or have large amounts of time and patience to invest in learning. Flexible learning environments such as Perseus may be designed primarily to support self-directed learning but also must provide mechanisms such as the path tool that allow students to follow carefully designed instructional paths.

Rather than simple variety of activities, the fact that mixed assignments began with directed and then allowed self-directed exploration suggest that a staged progression of increasing learner control is superior to either the "cookbook" or "sink or swim" methods. This conclusion is due both to the psychology of learning and to the nature of the electronic medium. The former predicts that learner control is more engaging and thus more motivating for experienced learners. Moreover, new type of challenges in the form of assignments serves as stimulation for continued engagement--a self-reinforcing process. The latter predicts that learning within a complex system is distinct from using it to accomplish tasks and that progressive disclosure or minimalist instruction may be the best way to apply such systems (e.g., Carroll, 1990). Since systems such as Perseus are highly complex to learn to use and to actually use after system learning has taken place (there are a multitude of features and alternative tools for accomplishing any task), an incremental experience can minimalize overload and frustration. Taken together, these two notions imply that if hypermedia systems are to be effective for learning, they should be applied for sustained periods with varied assignments that are staged from directed to undirected progression. Variety is the spice of learning. However, the tension between learner directed and teacher directed learning is best managed by insuring that the variety is not random.

References

Carroll, J. (1990). *The Nurnberg funnel: Designing minimalist instruction for practical computer skill.* Cambridge, MA: MIT Press.

Crane, G. (1988). Redefining the book: Some preliminary problems. *Academic Computing*, 2(5), 6-11, 36-41

Crane, G. & Mylonas, E. (1988). The Perseus Project: An interactive curriculum on classical Greek civilization. *Educational Technology*, 28(11), 25-32.

Duffy, T. & Knuth, R. (1989). Hypermedia and instruction: Where is the match? *Designing Hypermedia for Learning.* D. Jonassen and H. Mandl (Eds.) Berlin: Springer-Verlag. 99-225.

Flagg, B. (1990). *Formative evaluation for educational technologies.* Hillsdale, NJ: Lawrence Erlbaum.

Kinzie, M. & Berdel, R. (1990). Design and use of hypermedia systems. *Educational Technology Research & Development*, 38(3), 61-68.

Marchionini, G. Evaluating Hypermedia-based Learning. *Designing Hypermedia for Learning.* D. Jonassen and H. Mandl (Eds.) Berlin: Springer-Verlag. 353-373

Marchionini, G. & Crane, G. (1994). Evaluating hypermedia and learning: Methods and results from the Perseus Project. *ACM Transactions on Information Systems*, 12(1), 5-34.

Mager, R. (1964). Learner-controlled instruction--1958-1964. *Programmed Instruction*, 4(2), 1-12.

Morrell, K., Marchionini, G. & Neuman, D. (1993). Sailing Perseus: Instructional strategies for hypermedia in the classics. *Journal of Educational Multimedia and Hypermedia*, 2(4), 337-353.

Mylonas, E. (1992). An interface to classical Greek civilization. *Journal of the American Society for Information Science*, 43(2), 192-201.

Neuman, D. (1991). Evaluating evolution: Naturalistic inquiry and the Perseus Project. *Computers and the Humanities*, 25, 239-246.

Ross, S. & Morrison, G. (1989). In search of a happy medium in instructional technology research: Issues concerning external validity, media replications, and learner control. *Educational Technology Research & Development*, 37(1), 19-33.

Hypermedia Courseware and Collaborative Tools for MS-Windows Environment

JOAQUIM ARNALDO MARTINS & JOAQUIM SOUSA PINTO
University of Aveiro / INESC
Campo Universitário, 3800 Aveiro, Portugal
Email: jam@inesca.pt

Abstract: This paper describes the design of a set of hypermedia tools intended to produce courseware, to be used for self learning or distance learning and training environments in a cooperative way. These tools has roots in an standalone hypermedia editor developed to create courseware, to which where added other tools to enhance it's usabilty, namely an hypermedia player to view courseware in self study mode or in groups, an hypermedia browser to create, to navigate or to display the hypermedia document structure and some tools to allow cooperative work. The underlying metaphors are the overhead projector, the lesson and the transparency. This system was developed for the MS-Windows environment, supporting OLE technology.

Nowadays the learning process is not confined to the childhood and school. Professional training, retraining and specialization involves a large number of people. The geographical spreading of these people raises the necessity to have distributed learning environments supported by intelligent tutoring systems were individuals or organizations can access to get self-learning information. Hypermedia systems (Nielsen, 1990) have enormous potential for training and education in scientific and non-scientific areas.

Rather than individual or self-learning it is important to create central learning and training centers that learners can access for learning and group study. Bearing this in mind, we are trying to setup some experiencies for cooperative group study and teaching, persuading the teachers to produce their teaching material in hypermedia support, instead of the traditional material (transparencies and lithographed material). Achieved this point, will be possible to the students to access the lesson's material, starting self-study or group study and, with the teachers support, to create lecture sessions. To support this environment we developed some hypermedia tools, able to handle several kinds of multimedia objects (text, sound, images and video) to produce, to organize and to play courseware in standalone or in group mode, as well as some underlying architecture to support these computer based cooperative activities.

Since we used the *OLE technology* to develop the described hypermedia tools we will introduce it's basic concepts. The central philosophy behind OLE technology is to change, from a traditional application centered view of computing, towards a document centered view (Microsoft, 1992). This implies that a user does not have to leave an application to create or edit application specific data but instead, more comfortably, simply launch an application with a mouse selection on the specific data. In the OLE scheme, applications can either be clients or servers. *Server applications* are programs where linked or embedded data is created. *Client applications* are programs that receive OLE data that is linked or embedded. Each OLE object has a graphical representation and a behavior. The OLE client controls the OLE object graphical representation and is responsible for the shape assumed by the object in the document. The OLE Server handles the OLE object behavior through its *verbs*.

Underlying Collaborative Architecture

In centralized training and tele-teaching systems there is a conference server that deals with the social rules and with the database needs (Derycke, 1993). In such a system, the participants in a conference should know, in advance, what is its server location and what is the local library. People during a conference can not access to other libraries nor can meet at different places. For big conferences this scenario fits well, but for small conferences where it is needed some flexibility, it should be possible for the various participants to decide about the best room to meet and to access to all documents of each participant. Each place must be able to host at any time a conference and so it must support a series of tools to help in the organization, preparation and execution of these meetings. Such a distributed system as some advantages, namely: the independence of a specific machine or network connection (it is possible to choose another place to meet) and the possibility of having several conferences simultaneously over the same network, providing that groups are independent. The disadvantage is that the client must be prepared to play also the *server role* (when selected as a host) and must deal with the problem of replicated objects which belong to a group of people in an efficient way (Abdel, 1988), since each user must have a copy of the working document.

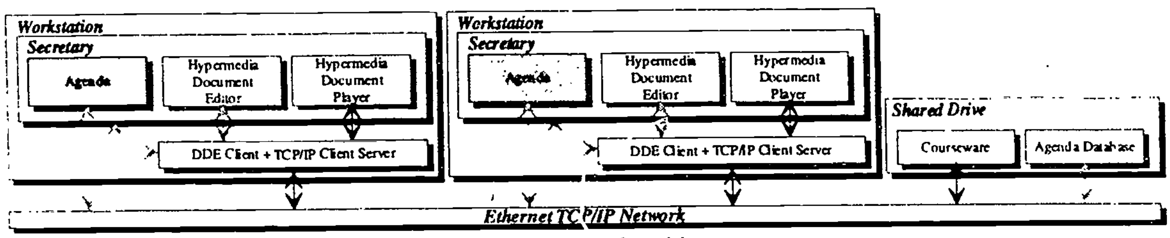


Figure 1. Proposed architecture

To implement the scenario shown in Figure 1, we used the MS-Windows environment taking profit of the OLE technology. For now, we built the following group of tools: *Hypermedia Editor*, *Hypermedia Player* and *Hypermedia Browser*. The Hypermedia Editor was designed to produce and annotate the courseware, for now in stand-alone mode. The Hypermedia Player was designed for self-study, learning and group study. The Hypermedia Browser works jointly with the Editor and Player and may be used to browse the lessons or to build the hypermedia str- -ure. *Secretary* and *Agenda* tools will be the engine for the proposed architecture. *Shared Drive* is implemented through a MS-DOS file system extension (Novell Netware, NFS, Windows for Workgroups, etc).

The main difficulty to design a Cooperative Hypermedia Editor, Browser or Player for this environment concerns with the granularity imposed by OLE technology. It is confined to the object level since we can not have the control of OLE servers. This is not a severe problem since are the authors that control the level of object granularity and from the pedagogical point of view, the transparencies should present clear ideas and key points without being overcrowded with information, which may be confusing. They should use simple objects without to much complexity, which is consistent with the metaphor of a transparency used to design the system.

Hypermedia Editor, Browser and Player Metaphors

A full technical description of the *Hypermedia Editor* describing the characteristics and functionality's can be seen in Pinto (1994). For an ergonomic analysis and a detailed description of the advantages and disadvantages of the use of OLE technology in hypermedia see Pauwels (1993, 1994). Just to understand the metaphors used in the design of the hypermedia engine they will be described, briefly, in the following paragraphs.

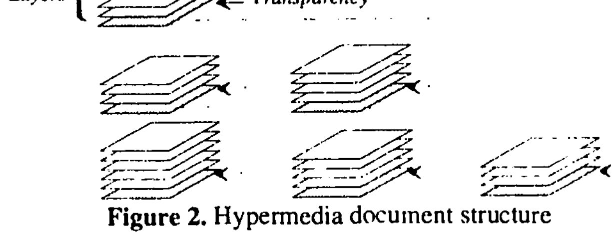


Figure 2. Hypermedia document structure

The metaphors used to describe the applications are the *hypermedia overhead projector* on which *lessons*, composed of piles of *hypermedia transparencies*, can be displayed either in a simple sequential order or in a more sophisticated one, so called "web" structure, as shown in Figure 2. One hypermedia node, consists of a *transparency* which may, or may not, be covered by one or more *overlaying layers*. A transparency is composed by *multimedia objects* and linked through *anchors* with other transparencies or layers. A transparency yields anchors and multimedia objects; the layers only hold multimedia objects.

The hypermedia engine admits links between transparencies, layers and other lessons. There is no need of a script language to create or to control the engine since the transparency and layer insertion and the anchoring among them is constructed automatically through the user actions.

An anchor establishes a link among a source and a destination point. The result of selecting an anchor is to follow up the link to its destination, that may be another transparency, a *superimposed layer*, an *alternated layer* or even another lesson. A multimedia object is any embedded or linked OLE object and may be time-variant or change its value. It is therefore a *hyperlink* to the associated server, which implies that an object activation may result in a video sequence presentation.

Hypermedia Browser

In very complex hypermedia lessons it is easy for a learner to get "lost" inside the lesson. Therefore it is necessary to have a structure visualization to help the user to navigate through, or browse, the lesson (Dumais, 1988; Nielsen, 1990). Such tool may also be used as a cooperative authoring tool used by several teachers to define the structure of the lessons; afterwards each teacher can edit his set of transparencies. The granularity level is confined to the object level, in editing mode, since to edit an object the control passes to the OLE server application which puts some problems to work in a cooperative way. The authors can cooperate on the lesson structure definition and object placement but they will be able only to edit single objects in an individual way, which we do not see as a restriction. The *Hypermedia Browser* presents the lesson structure independently of its content. Figure 3 shows the interaction between the Browser, the Editor and the Player, in stand-alone mode. In the Browser window, each transparency is identified by an icon and its name, and the connections among transparencies are represented by arrows.

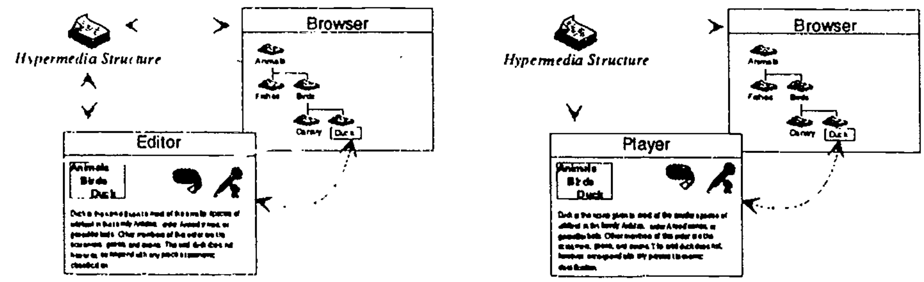


Figure 3. Interaction between the Browser, the Editor and the Player

To present the lesson structure, the browser linearizes and places it in a *virtual grid* using an algorithm that recalculates the lesson network each time its structure changes. The links in the network are calculated using the anchors among transparencies as reference. Some paths may be difficult to identify in intricate structures. To improve this and to allow a better understanding of the network, the browser draws the links in eight colors, that users may select or deselect. Besides the color mechanism, a user can always know the entire list of connections and the associated layers of a transparency simply through a mouse selection on it.

Using the Browser to build lessons

Any user, without programming skills can construct complex lessons, because the process is entirely done visually. A user inserts new blank transparencies and then anchors them with each other through a dialogue box that shows the possible link destinations. Each time a user inserts a new anchor among transparencies, the Browser recalculates the network. Depending on the depth of the source transparency in the network, it places a visible anchor on left or on the right side of the source transparency. Usually the network has bi-directional links. To simplify the presentation and make it more readable, bi-directional links are drawn through a bi-directional line.

Using the Browser to browse information

The other Browser functionality is to allow the user to browse through the lesson web structure. The currently displayed transparency on the overhead projector is identified in the Browser window by a shaded look. To jump into another transparency in the Editor or Player, a user should simply, in the Browser window, click over the desired transparency. The Browser sends an order to the Editor or Player to display the selected transparency. This behavior is very useful if the user is "lost" in the structure and wants to go to a "known" point.

Hypermedia Player

To play the lessons it is not needed a full Editor. The *Hypermedia Player* tool allows the students to browse documents in self learning mode or, in a cooperative environment, controlled by a teacher or a group moderator.

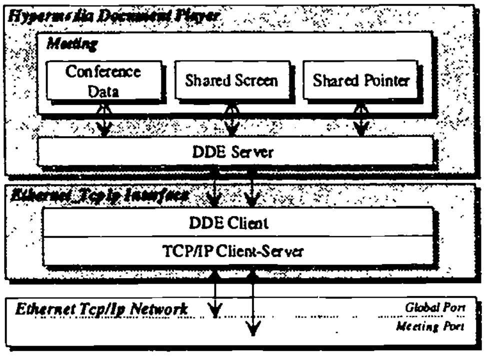


Figure 4. Hypermedia Player Internal Architecture

In the Player, the students are not able to modify or annotate the lessons and have a simple mechanism to go through the transparencies.

The Cooperative Player

The architecture for cooperation between the students and teachers is built on top of a TCP/IP network. The communication structure is not based on the client/server classical model. Instead, any participant has the possibility to host a meeting, i.e., each station is simultaneously client and server. This means that does not exist any global server able to deal with the global data, such as the name of all running meetings. Each meeting has a *proponent* that is responsible for the *meeting data* and to play the server role. The meeting data has three components: *Conference Data*, *Shared Screen* and *Shared Pointer*. The Conference Data includes the name of all running conferences and when the user is attached to a specific one, the name of all elements in the conference. The Shared Screen holds the size and scroll bar positions of the public overhead projector. The Shared Pointer section has the queue of users for the shared pointer, the shared pointer position and the identification of the actual owner, if any.

Duė to the non existence of a central server to provide information about meetings, we created a *global channel* listened by all users. Always a user plans to connect to an existent meeting, sends a network message, through the global channel, requesting information about running meetings. Each proponent replies with the meeting name and with the *meeting channel*, that is the channel through which the conference specific data flows.

After a user connects to a meeting, the proponent synchronizes the new station. This involves the new user local screen sizing, because all partners should see the same thing at the same time (WYSIWIS), and the delivering of all conference data. Each user can have a request for the global pointer possession in the global pointer queue, that follows the "first come first goes" algorithm. The proponent has the privilege to change the user's order in that queue and revoke the pointer possession from a user. The actions performed by the global pointer owner, such as a OLE object activation or a window scrooling, are reflected in all stations in a meeting.

A user, unless is the global pointer owner, can leave the meeting at any time. The meeting finishes when the proponent leaves the meeting.

Figure 5 shows the look and feel of the hypermedia editor, browser and player.

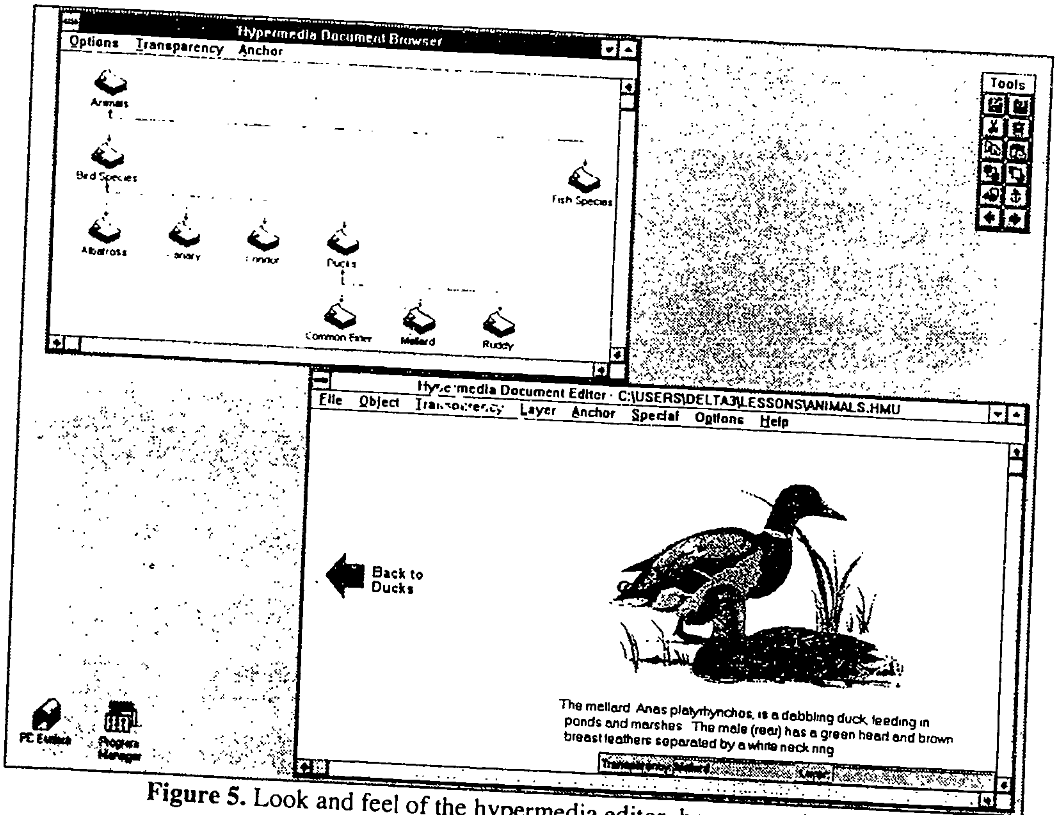


Figure 5. Look and feel of the hypermedia editor, browser and player

Future Work on the Distributed Cooperative Environment

Now we discuss some ideas to enhance the work in a cooperative way. Some of them are already on going work, while others are ideas we starting to explore, namely the transmission of sound and image over TCP/IP protocols in a synchronous and asynchronous way, which will improve the quality of computer supported collaborat e work over TCP/IP networks.

Associated with this cooperative environment it is needed to have special attention to aspects like document distribution and storage and to have some tools to make these systems more usable and ergonomic.

Collaborative Tools

Some of the tools needed in our environment are already used in other collaborative environments (Derycke, 1993), namely a *meeting room*, a *vote* mechanism to help to take decisions, a *timer* to control the duration of talks, pointer possessions and meetings. Mary, a blond blue eyes *secretary*, an *agenda*, a *briefcase* to carry personal documents to the meetings, a *synchronous sound server* to distribute sound/voice to all participants and an *asynchronous sound and video server* are other tools included into our environment.

Under current implementation there are the sound and image servers, the agenda and the voting mechanism. Sound and video are carried over IP networks and the servers are software based and will control the conversation among a small group of people. Some tests done in our TCP/IP/Ethernet network, and using a Sound Blaster board as the input and output device for voice, shown that until a network load of 60-70%, depending on several configuration factors, the users don't detect any degradation in the voice signal for packet losses of 2% (Azevedo, 1994). Now we are studying some form of asynchronous communication with voice and even appended video, based on the fact that in some scenarios the conversation is done in message bursts.

In the *Agenda* the users can see a list of booked meetings, create new meetings, adhere to a meeting or to see meeting details, such as the date and time, participant names, requirements, etc. It is possible to create public meetings, in which all users can participate, and private meetings. In this case, only the participants included in the participants list are allowed to participate or see the meeting details. The meeting requirements contain a list of the meeting files, that may be placed by Mary in each user briefcase before the meeting start.

Besides the Agenda, it is possible in any moment during a meeting to admit new users to the group. In this case, the meeting proponent will be responsible to admit or not admit the new user to the meeting or to submit the problem to all members of the group by a voting process. This process is done using the *Vote* tool. The vote process may be open or secret. In first case, each participant knows the vote of each other and the result is done through the difference between the votes. In the last case the participants only know the result. This an example of the utility of the vote tool that may be used in many other situations during the meeting.

Document distribution and storage

Hypermedia documents may have an enormous amount of data, which puts several problems such as large databases and long transmission times. It will be useful to use compression and pre-fetching techniques to reduce storage space and distribution time. The pre-fetching can be used to distribute documents before the meeting. The *Secretary* looks up to the *Agenda* and sometime before the meeting start, fits the boss *briefcase* with the conference stuff. During the meeting the pre-fetching technique may be used to load the next possible transparencies to minimize the time between transparencies. It is also important to look for distributed storage systems, since there are documents spread over several rooms and libraries. But to have organized libraries of multimedia courseware and to manipulate and access them in an efficient way it is needed to look for distributed multimedia databases and high level APIs able to handle different types of databases and/or file systems.

Conclusions

We described a set of hypermedia tools implemented under the MS-Windows environment to browse and play hypermedia documents in standalone or in a cooperative way, using the OLE technology, which open the possibility to integrate into the hypermedia documents, objects produced by any MS-Windows OLE server application. This allows the users to use their familiar applications to produce, annotate or play the courseware, which, jointly with the adoption of overhead projector and transparency metaphor's, results in a simple and ergonomic user interface.

References

Abdel-Wahab, H.M. & Guan, S. & Nievergelt, J., (1988). "Shared Workspaces for Group Collaboration: An Experiment Using Internet and UNIX Interprocess Communications", in *IEEE Communications Magazine*, November, 10-16

Azevedo, J. & Martins, J.A.& Oliveira, P.G. (1994). "An Experimenta Study on Voice/Data Integration over an Ethernet LAN", in *Proccedings of INDC'94 - International Conference on Information Networks and Data Communications*, April 18-21, Madeira, Portugal.

Derycke, A.C. & Kaye, A.R (1993). "Participative modelling and design of collaborative distance learning tools *Proceedings of the IFIP TC3 Third Teleteaching Conference*, 191-200.

in CO-LEARN project", in *Proceedings of the IFIP TC3 Third Teleteaching Conference*, 191-200.

Dumais, S., (1988). "Handbook of Human Computer Interaction", in *Textual Information Retrieval*, Ed. Martin Helander, North-Holland.

Microsoft Press (1992), "Object Linking and Embedding - Programmer's Reference".

Nielsen, J.(1990). "Hypertext and Hypermedia", *Academic Press.*

Pauwels, H.W.J.B. & Pinto, J. S. & Santos, B. S. & Martins, J. A. (1993). "An Integrated Courseware Editor Based on OLE Technology", in *Lectures Notes In Computer Science*, 733, 439-440.

Pauwels, H.W.J.B. & Pinto, J. S. & Santos, B. S. & Martins, J. A. (1994). "Integrating Existing Applications in Hypermedia Learning Material (General Issues & Experiences with OLE technology)", in *Proccedings of ED-MEDIA 94 - World Conference on Educational Multimedia and Hypermedia*, June 25-29, Vancouver, Canada.

Pinto, J. S. & Pauwels, H.W.J.B. & Martins, J. A. & Santos, B. S. (1994). "HyDE: a hypermedia document editor based on OLE technology", in *Proccedings of ICMCS'94 - International Conference on Multimedia Computing*, May-15-19, Boston, MA, U.S.A..

Acknowledgements

We would like to thank Hubert B. Pauwels and Beatriz S. Santos who actively collaborated in the design of the standalone document editor. This editor was also funded by the european DELTA project CO-LEARN (D2005).

VideoScheme: A Research, Authoring, and Teaching Tool for Multimedia

J. MATTHEWS, F. MAKEDON, AND P. GLOOR
Department of Mathematics and Computer Science
Dartmouth College, Hanover, NH, 03755, USA
E-Mail: James.W.Matthews@dartmouth.edu

Abstract: The availability of digital multimedia technology poses new challenges to researchers, authors, and educators, even as it creates new opportunities for rich communication. This paper suggests interactive computer programming as a fruitful approach to these challenges. VideoScheme, a prototype video programming environment, is described along with promising applications.

Introduction

The advent of affordable digital audio and video has unleashed new possibilities for interactive multimedia.[3] But it has also raised a host of questions, of how best to acquire, analyze, edit, and deliver digital media products. To answer these questions researchers and practitioners need new tools, tools with enough flexibility to accommodate new demands, explore original hypotheses, and prototype novel applications. We believe that VideoScheme, a programming environment for digital media, offers a valuable example of such a tool.[4]

System Description

VideoScheme is implemented as an application for the Apple Macintosh, written in C and totaling approximately 100KB of executable code. It provides a visual browser for viewing and listening to digital movies, using Apple's QuickTime system software for movie storage and decompression.[9] The browser displays video and audio tracks in a time-line fashion, at various levels of temporal detail. Clicking on the video tracks displays individual frames in a "flip book" fashion, while clicking on the audio track plays it back; clicking twice in rapid succession plays back both audio and video.

As a visual interface to digital media VideoScheme is nothing out of the ordinary; what separates it from conventional computer-based video editors is its programming environment. VideoScheme includes an interpreter for the LISP-dialect Scheme, built on the SIOD (Scheme-in-one-Defun) implementation, along with text windows for editing and executing Scheme functions.[10] Functions typed into the text windows can be immediately selected and evaluated. The environment, while deficient in debugging facilities, offers such standard LISP/Scheme programming features as garbage collection and a context-sensitive editor (for parentheses matching). In addition it offers a full complement of arithmetic functions for dynamically-sized arrays, an important feature for handling digital video and audio.

Scheme was chosen over other alternatives (such as Tcl, Pascal, and HyperTalk) for a number of reasons. Scheme treats functions as first class objects, so they can be passed as arguments to other functions. This makes it easier to compose new functions out of existing ones, and adds greatly to the expressive power of the language. Scheme is also easily interpreted, a benefit for rapid prototyping. Scheme includes vector data types, which map very naturally to the basic data types of digital multimedia, namely pixel maps and audio samples. Finally, Scheme is easily implemented in a small amount of portable code, an advantage for research use. The most significant drawback to using Scheme is the programming syntax, which non-programmers (and even some programmers) find difficult to use. If we were to start again we might consider Logo or Dylan, languages with the positive properties of Scheme but with more attractive syntax.

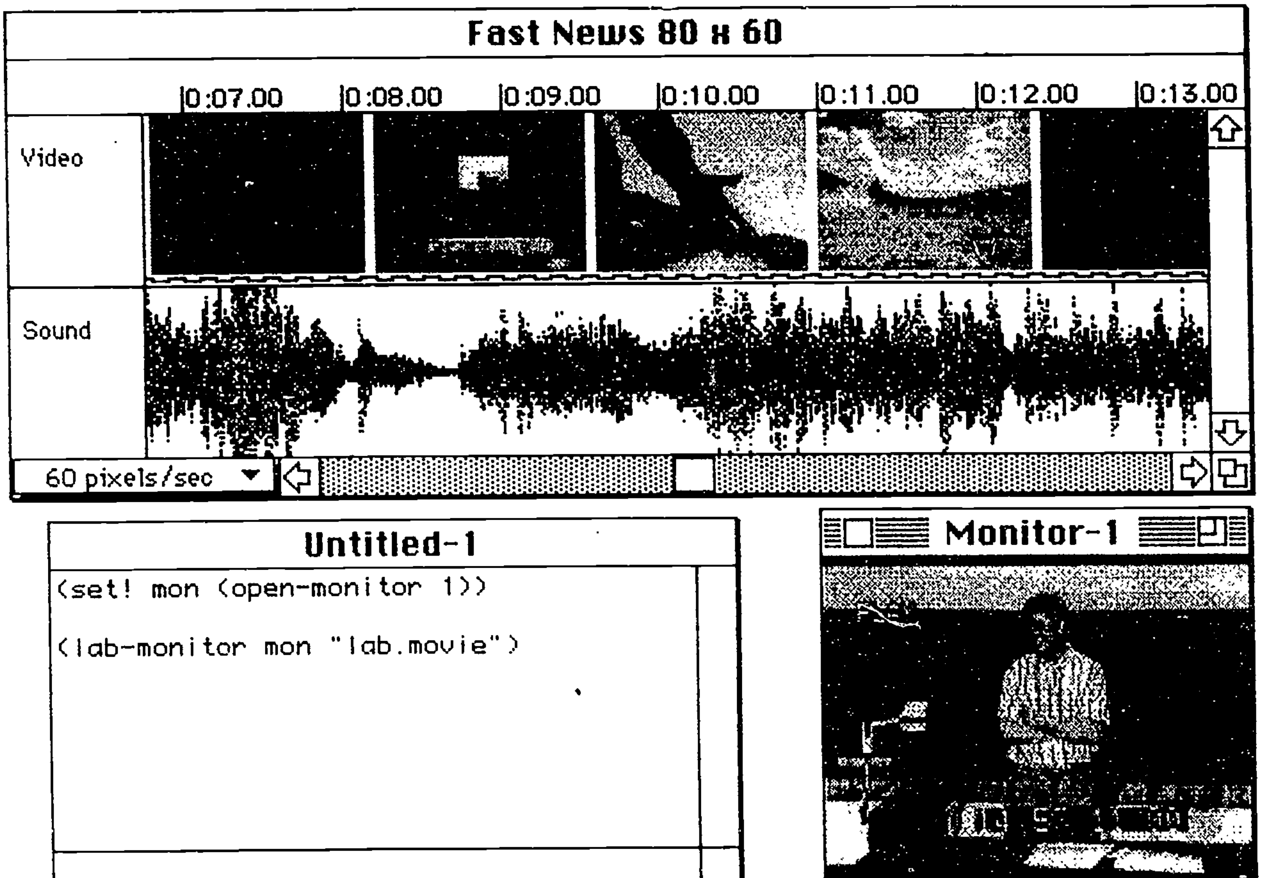


Figure 1 VideoScheme user interface, with movie (top), monitor (right), and program editor (left).

Language Description

VideoScheme extends the Scheme language to accommodate digital media. In addition to the standard number, string, list, and array data types VideoScheme supports the following objects:

movie	-- a stored digital movie, with one or more tracks.
track	-- a time-ordered sequence of digital audio, video, or other media.
monitor	-- a digital video source, such as a camera, TV tuner, or videotape player.
image	-- an array of pixel values, either 24-bit RGB or 8-bit gray level values.
sample	-- an array of 8-bit Pulse Code Modulation audio data.

These objects are manipulated by new built-in functions. Movies can be created, opened, edited, and recorded:

```
(new-movie)                    ; creates and returns a reference to a new movie

(open-movie filename)          ; opens a stored movie file

(cut-movie-clip movie time duration)
                               ; moves a movie segment to the system clipboard
(copy-movie-clip movie time duration)
                               ; copies a movie segment to the system clipboard
(past-movie-clip movie time duration)
                               ; replaces a movie segment with the segment on the clipboard
(delete-track movie trackno)
                               ; removes a movie track
```

```
(copy-track movie trackno target)
                                  ; copies a movie track to another movie
(record-segment monitor filename duration)
                                  ; records a segment of live video from the monitor to a file
```

Images and sound samples can be extracted from movie tracks or monitors, and manipulated with standard array functions:

```
(get-video-frame movie trackno time image)
                                  ; extracts a frame from a video track
(get-monitor-image monitor image)
                                  ; copies the current frame from a video source
(get-audio-samples movie trackno time duration samples)
                                  ; extracts sound samples from an audio track
```

With this small set of primitive objects, and small number of built-in functions, we can rapidly build a wide variety of useful functions with applications in research, authoring and education.

Research Applications

With powerful computers and digital media it is possible to build systems that perform automatic analysis of video and audio data. The results of this analysis can be used to make existing applications (such as indexing, retrieval, and editing) more efficient, or to enable entirely new applications.[6, 11, 12] It would certainly be desirable if computers could perform these analytical tasks as well as a trained human; but short of such a breakthrough in artificial intelligence there are numerous less-ambitious goals to be pursued by researchers in this field.

One important goal is an automatic system for dividing digital video into segments, by detecting the "cuts" that divide one shot from another. There is no perfect definition of a "cut," but generally the term refers to a sharp discontinuity in a video stream, such as the break between two recording bursts in unprocessed video, or the point where two clips were concatenated in the editing process. Once cuts are found the segments they define can be represented by a subset of the segment's data (e.g. the first and last frames), since the continuity of the segment ensures a great deal of information redundancy. This reduction of a potentially long segment to a few frames is a significant boon to a number of applications, such as indexing, logging, navigation, and editing, since those tasks may be performed on a greatly reduced set of data.

A number of algorithms have been proposed for automatic cut detection, and one of the advantages of VideoScheme is that such algorithms can be implemented as compactly as their mathematical formulation. For example, a simple measure of visual continuity is a sum of pointwise differences in gray value or color. Such a test can be performed by the following fragment of VideoScheme code:

```
(adiff frame1 frame2 delta)     ; subtract arrays of gray value
(aabs delta)                    ; compute absolute difference values
(atotal delta)                  ; sum differences
```

Scene changes trigger large pointwise differences, but this measure is also very sensitive to camera motion and zooming, which may change every pixel without introducing a new scene. Refinements of this algorithm, such as one that counts the differences that exceed a threshold, have a similar weakness. So there appears to be some value in a test that is not so spatially sensitive, such as the difference between summed gray values:

```
(- (atotal frame1) (atotal frame2))  ; subtract summed gray values
```

This measure is insensitive to camera pans and zooms, but it is also insensitive to actual cuts, since the average gray level may not change dramatically across the cut. A more reliable indicator is the gray level or color histogram. Using VideoScheme's built in color histogram function we can easily compute this measure:

```
(get-color-histogram64 frame1 histogram1) ; compute 64-bucket color histograms
(get-color-histogram64 frame2 histogram2)
(adiff histogram1 histogram2 delta)       ; subtract histograms
(aabs delta)                              ; compute absolute differences
(atotal delta)                            ; sum differences
```

This test can be further refined, by breaking each frame into a number of sub-frames and discarding the ones with above-median changes, or counting the number of changes that exceed a threshold; either modification is quickly implemented in VideoScheme, and each makes the algorithm more robust against local phenomena such as object motion.

While histogram comparison is widely considered a robust solution to detection of simple camera breaks, the general problem remains a fertile area for new approaches. In recent months novel algorithms have been proposed for detecting gradual transitions, and for using motion-sensitive measures in conjunction with a projection detecting filter.[12, 7] The future is bound to bring still more proposed algorithms, targeted at specific video sources and applications. We believe that VideoScheme's flexibility makes it a useful vehicle for this research. VideoScheme's high-level primitives and rapid-turnaround programming environment make programs compact and prototyping very rapid. Nagasaka and Tanaka's histogram-based algorithm has been implemented as 25 lines of VideoScheme, and the Otsuji-Tonomura filter was implemented in a couple of hours.[6, 7] We aim to implement and evaluate new cut detection algorithms as they are proposed, and hopefully develop our own improvements.

Another area of research is the application of automatic video analysis to interactive multimedia. Linblad has termed this "computer-participative multimedia" since the computer is actively involved, reacting to the content of the video data.[2] VideoScheme's monitor object makes it possible to easily implement such applications. For example, this VideoScheme code fragment implements the core of a video room monitor:

```
(set! camera (open-monitor 1))           ; create a monitor object for a camera
(get-monitor-image camera baseline)      ; capture a baseline image
(while t
 (get-monitor-image camera new-image)              ; capture a new image
  (if (image-diff baseline new-image)              ; compare to baseline
   (record-segment camera "monitor-movie" 5.0)))   ; record 5 seconds of video
```

With a camera pointed at a door this fragment keeps a running log of all the people who enter and leave the room. With the addition of a sound playback command this function could also serve as a video answering machine: people approaching a closed door could be automatically prompted to leave a videotaped message. This fragment assumes that any significant change in the image striking the camera represents a person. More sophisticated analysis functions could look for more specific phenomena. So, for example, they could automatically record television programs with a certain opening screen or on-screen logo.

Authoring with Programming

Digital media technology has enabled more than the creation of movies on computers; it has also made possible new sorts of information systems. One example is the *Parallel Computation*, a CD-ROM that integrates digital video, audio, animation, and hypertext to communicate the proceedings of an academic conference.[5] The unusual nature of the system posed numerous challenges to the development staff, challenges that were only partly addressed by existing authoring tools. For example, the centerpiece of the proceedings is 8 forty-five minute synchronized audio/video presentation of speech with overhead slides. It was sometimes necessary to move audio tracks independently of the accompanying video, to accommodate sound editing software. Our commercial tools did not provide a way to do this without re-compressing six hours of animations, but VideoScheme was quickly adapted to the task. Earlier in the process we needed to remove silences from the original 12 hours of sound tracks, a painstaking manual process. Had VideoScheme been available at that time it would have been a simple function:

```
(while (< time (get movie-duration movie)) ; loop through whole movie
 (if (silence? movie trackno time 0.1)      ; if there is 0.1 seconds of silence
  (cut-movie-clip movie time 0.1)           ; then remove that segment
  (set! time (+ time 0.1))))                ; otherwise move on to next segment
```

We also removed noise words (such as "um" and "ahh") by hand, and we believe that VideoScheme could be extended to assist in this step as well.

It was clear from our experience that while existing tools offer high-quality solutions for problems in their domain, they are often poorly suited to the ad hoc tasks that arise in innovative multimedia authoring. In such cases a programmable system may provide the necessary flexibility to turn a tedious, manual process into an automatic operation. It may also permit entirely novel operations. In his work on a programmable graphics editor Eisenberg noted that such an editor can produce effects, such as fractals and recursive Escher-like designs, that would be near-impossible with manual tools.[1] Likewise we can imagine VideoScheme programs to implement algorithmically defined effects, such as fades and wipes, that could hardly be achieved any other way.

A critical concern with authoring tools is the speed with which new applications can be developed. Traditional programming environments (featuring compiled languages such as C and C++) offer complete access to the computer hardware and data, but can require months or years of effort to produce a usable tool. Object-oriented frameworks accelerate the process somewhat, but even then many details must be coded by hand. By contrast, a rapid-turnaround interpreted language like VideoScheme makes it possible to write applications in minutes or hours. Tool authors do not need to master the low-level details, and powerful operations can be composed out of very high-level primitives and previously-written functions.

Education in Multimedia

Education is an obvious application for multimedia, but it is also true that students will need education **in** multimedia. As more data becomes available in multimedia form it will become more important for students to understand the fundamentals of the technology, in order to use this form of communication and to be discerning consumers of multimedia information. Commercial multimedia tool developers are one source of teaching tools, since their products are typically very polished and focused on the task of making multimedia production as easy as possible.[8] But there is also a call for more basic, flexible tools, that expose more of the underlying technology and permit easy experimentation with features that might not be included in a commercial package.

We believe that VideoScheme is such a tool, and to that end made it the subject of a project in a multimedia seminar for undergraduates and graduate students in computer science. All the students had programming experience, but many had not used Scheme or a LISP dialect and none had written programs to manipulate digital video. The lack of debugging facilities in VideoScheme was a troublesome obstacle, as was the parentheses-heavy syntax of Scheme. Nonetheless, after a 90-minute tutorial most students were able to write functioning VideoScheme programs to perform a specified editing task and implement a crude cut detection algorithm. They were also able to informally analyze the performance of the cut-detection algorithm (which was based on an average gray-level test) and thereby make classroom discussion of its faults more informed and concrete. The flexibility of the system also prompted the students to suggest improvements and new applications, including the video answering machine mentioned above.

While learning to program is a significant obstacle for most students, a programmable multimedia environment appears to be a useful tool for exposing computer science students to digital media in a way that is concrete without unnecessarily restricting their experimentation. By creating multimedia programs students can gain appreciation for the ones they merely use, in the same way that writing fosters appreciation for literature. In the best of cases such an experience can help students to imagine and implement completely original applications without the expertise required by conventional low-level programming tools.

Conclusion

Digital media are producing exciting, and rapid changes in the way we communicate. There is much research to be done, and much that the multimedia producers and consumers of the future should learn, in order to fully exploit the potential of this technology. We believe that programmable environments such as VideoScheme provide the flexibility to meet this dynamic challenge, and we look forward to finding new applications for this system.

[1] Eisenberg, M. "Programmable Applications: Interpreter Meets Interface." MIT Artificial Intelligence Memo 1325, October 1991.

[2] Linblad, C. "The VuSystem: A Computer-Participative Multimedia Toolkit." MIT Laboratory for Computer Science, 1993.

[2] Linblad, C., Wetherall, D., and Tennenhouse, D. "The VuSystem: A Programming System for Visual Processing of Digital Video." TR-590, MIT Laboratory for Computer Science, 1993.

[3] Mackay, W. and Davenport, G. "Virtual Video Editing in Interactive Multimedia Applications." Communications of the ACM, 32:7, July 1989.

[4] Matthews, J., Gloor, P. and Makedon, F. "VideoScheme: A Programmable Video Editing System for Automation and Media Recognition." ACM Multimedia '93, Anaheim, CA, 1993.

[5] Metaxas, T., Cheyney, M., Gloor, P., Johnson, D., Makedon, F., and Matthews, J. "Conference on a Disk: An Experiment in Hypermedia Publishing." ED-MEDIA '94, Vancouver, Canada, 1994.

[6] Nagasaka, A. and Tanaka, Y. "Automatic Video Indexing and Full-Video Search for Object Appearances." IFIP Transactions A (Computer Science and Technology), vol. A-7, 1992.

[7] Otsuji, K. and Tonomura, Y. "Projection Detecting Filter for Video Cut Detection." ACM Multimedia '93, Anaheim, CA, 1993.

[8] Premiere. Adobe Systems Incorporated. Mountain View, CA.

[9] QuickTime. Apple Computer, Inc. Cupertino, CA.

[10] SIOD (Scheme-in-one-Defun). Paradigm Associates, Inc. Cambridge, MA.

[11] Ueda, H., Miyatake, T., and Yoshizawa, S. "IMPACT: An Interactive Natural-Motion-Picture Dedicated Multimedia Authoring System." CHI '91 Conference Proceedings, 1991.

[12] Zhang, H., Kankanhalli, A. and Smoliar, S. "Automatic partitioning of full-motion video." Multimedia Systems, 1:1, August 1993.

History, Hypermedia and the Birth of a Nation

J. MESSING
School of Information Studies, Charles Sturt University-Riverina,
Locked Bag 675, Wagga Wagga, 2678, Australia
E-Mail: jmessing@csu.edu.au

R. McLACHLAN
School of Communication and Liberal Studies, Charles Sturt University-Mitchell,
Private Bag 7, Bathurst, 2795, Australia

Abstract: The Gallipoli project brought together materials from a national war archive to develop a multimedia teaching resource in an area that is central to a great deal of both Australian culture and history. The material contained within these HyperCard stacks makes it possible for students to conduct serious research as well as giving them a feel for what many of the participants actually felt at the time. A variety of linking mechanisms, including a class based link, were employed to make the process of following various themes a manageable process.

Introduction

Gallipoli is a small peninsula at the north-eastern end of the Aegean Sea leading to the Sea of Marmara and Istanbul. The narrow passage between the Gallipoli peninsula and the Turkish mainland is known as the Dardanelles. However, to Australians, Gallipoli is more an event than a place. It represents as significant an event to Australians as does the storming of the Bastille to the French and the Boston Tea Party to citizens of the United States. It more accurately represents the birth of Australia as a nation than does its proclamation of independence. Indeed, ANZAC Day which commemorates the events that took place on Gallipoli on April 25, 1915 is conscientiously celebrated as a public holiday, more so than the official Australia Day (January 26) or even the anniversary of the date of Federation.

The events of Gallipoli form a most important part of Australian culture and consequently feature in the curriculum of every state educational authority in Australia. As an historical event during World War I, is was a notable failure and little more than a dot on the page of the events of 1914 - 18. Students learn of the courage, self sacrifice and other most admirable of human attributes that the soldiers of the first Australian New Zealand Army Corps (ANZAC) displayed during that campaign. However, what most students fail to ever gain from a traditional approach to the study of these events is a real understanding of the feelings of the soldiers, the deprivation, the futility or even an accurate representation of the events of the Gallipoli Campaign.

Australian history teachers have for quite some time been aware of the possibilities associated with the use of computer technology for the teaching of historical concepts. The development and dissemination of the First Fleet Database for the Apple II in 1982 was arguably, the alerting of many an Australian history teacher to these possibilities. Since that time there have been many other similar projects, each bringing something different, sometimes something better, to the use of computers as a resource in the teaching of history. The principal value of such databases has been that they permit students to explore the data, formulate their own hypotheses and test these within the confines of the data rather than being the passive receptors of someone else's interpretations. Both authors of this paper have been involved in such projects in various ways for nearly fifteen years.

While database management software has become more sophisticated and the types of queries which may be put, have also increased in sophistication, the inescapable conclusion that one comes to is that the actual software used has been very much a secondary consideration. The value of such databases lies in the richness of the data. While databases may have increased in size; both in the number of records and in the number of fields of data, thereby increasing the scope for meaningful learning, such data is still largely text. Consequently, it has not used the vast information carrying capacity that pictures, sound and movies possess which make such information available to a wider audience. Adding other media forms would make it multi-media but this is still a long way short of what is required in a true hypermedia system. True hypermedia provides a variety of access paths to any point in the data and will thus support a variety of learning activities and a range of people who may prefer to gain their information from sources other than text.

The Gallipoli stacks are a set of linked hypermedia documents written using HyperCard. They incorporate the full range of media; text, graphics, sound and video. They break new ground in the use of computers as a resource in the teaching of Australian history and while there are probably still many areas for improvement, they point to a new direction in serious historical reference material. They attempt to convey more than just historical accuracy. The vast resources of the Australian War Memorial were used to build an application which portrays events from many different perspectives including that of some of the actual participants as well as official accounts

A database of Australian soldiers was originally compiled using records submitted by students from all over Australia and implemented using Apple Works as part of the 75th anniversary of the Gallipoli campaign. It was from the discussions that the authors had concerning the conversion of this database to a more generally available computer platform that the Gallipoli stacks originated. There seemed little point in producing just another database and from the very outset the objectives of the project were widened to include creating a reference work which:

- supported a full range of media types
- would facilitate user browsing as well as direct inquiry
- contained a balance of material including soldiers accounts, official statistics and relevant factual data, a chronology of the events for the entire campaign not just the few days that are the most noteworthy and original material, some of which had never before been published.

What are the Gallipoli stacks?

These stacks are perhaps the best single collection of research and reference material relating to Australia's involvement in the Gallipoli campaign ever published. As well as Anzac letters, diaries and photographs from the Australian War Memorial, there are film clips, maps and the individual records of over 500 Anzacs. Daily accounts in the Diary stack provide an almost day to day chronology of the campaign and the Statistics stack contains a great deal of useful background information. These stacks provide a combination of a research archive and a reference library. They provide an assembled collection of information on the Gallipoli campaign, some of which schools would find extremely difficult if not impossible to obtain.

How can they be used?

This is largely up to teachers. They can be used to teach about the history of the Gallipoli campaign, to teach about conducting historical research or even examine issues such as bias in historical writings. One can simple browse through the stacks, reading letters and looking at photos, film clips etc with no particular goal in mind. Or one can use the facilities to track down answers to specific questions such as "What happened on April 25th?", "What did the Anzacs think of the Turks?", "What were the major battles?", "What did the Anzacs do in their spare time?", "How was the evacuation organised?" "What were conditions like on Gallipoli?" When ...? Where...? Why...? It would be presumptuous and restrictive for us to provide the questions! The stacks have not been designed to provide simple answers to prepared questions. To enter the Gallipoli stacks is not unlike entering the real archive or research library. Lots of help is provided but it is up to the researcher to formulate the questions and search for their answers. Teachers and students will find them useful; not just for the historical content but also in teaching researching and writing. Journalists and historians too, will find something of interest. Indeed, anyone with an interest in the Gallipoli campaign will find much in these stacks to further their understanding of this important chapter of our nation's history.

What makes the Gallipoli Stacks different?

There are a number of features which make this resource unique. One of these which sets this system apart from other multimedia or even other hypermedia systems is the range of linking that is possible between different parts of the system but especially from the diary. Obvious linking mechanisms from buttons and hypertext links from text are supported in a manner which greatly increases their usefulness. It soon becomes apparent to the user that the hypertext links are not all the same. There are three types of hypertext links which have been built in.

- Some text is linked to a single reference and in this case, clicking the text will take one directly to the reference.
- Some text has multiple references and in this case, clicking will display a list of all of these references and ask one to choose from the available list.
- Some text such as "Bully beef" may have direct references but it is also an example of a general category of "food".

Ten "class" or general category links have been constructed to permit a broader view and an exploration of various themes such as food, weather. When a list of references is displayed, the type of reference (Picture, Map, Text, Film, Database, Statistics) is also indicated.

An additional linking mechanism which is a little easier to use than HyperCard's Find command has also been created. This provides links to the next occurrence of a selected word or phrase from any part of the diary stack. This allows students to follow a theme using certain keywords.

Photographs and film clips have been digitised and compressed using QuickTime compression techniques which make such a multimedia information system feasible. While multimedia encyclopaedia are becoming more commonplace, they are still relatively scarce. This is the first attempt of an Australian historical subject which looks at it in such minute detail instead of the glossy but superficial treatment given by some other products. It is hoped that this type of information system is the vanguard for educational resources in the years to come.

What's in the stacks?

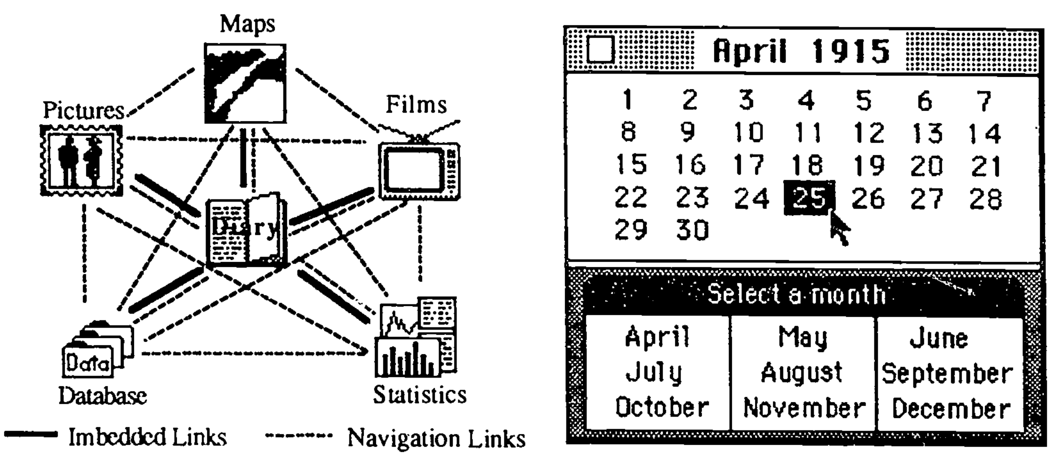


Figure 1.

The six elements of the Gallipoli material are shown in figure 1. Since the Dairy is the key to the system it is shown at the centre. Each of the stacks will work in its own right, and as shown, there are navigation links to take you from one to the others. However, the Dairy stack has many other embedded links which may be used to enrich the basic descriptions and discussions.

The Diary stack

This is organised like a daily diary that consists of two major panels of text which relate to events on that particular day. One contains extracts from letters and diaries which were written by soldiers. In most cases these accounts were written after the actual event. It is a little impractical to expect soldiers to stop fighting so that they can write a letter or make a diary entry!! In some cases these accounts were written when the person was wounded and recovering in hospital. Many diaries and letters held in the Australian War Memorial Archive were examined and representative samples selected. Several accounts were very complete and a number of these have been used throughout the stack to give a consistent point of view at different times. The second panel contains a description of the events of that day compiled by the authors. Sometimes this includes copies of official dispatches or signals that were sent (particularly on the days of major battles).

The actual landing at Gallipoli took place on April 25th and the last of the troops were evacuated in December 1915. The diary has entries for most days between April 1st and 31st December so that it covers all of the campaign as well as preliminary events which are of some significance.

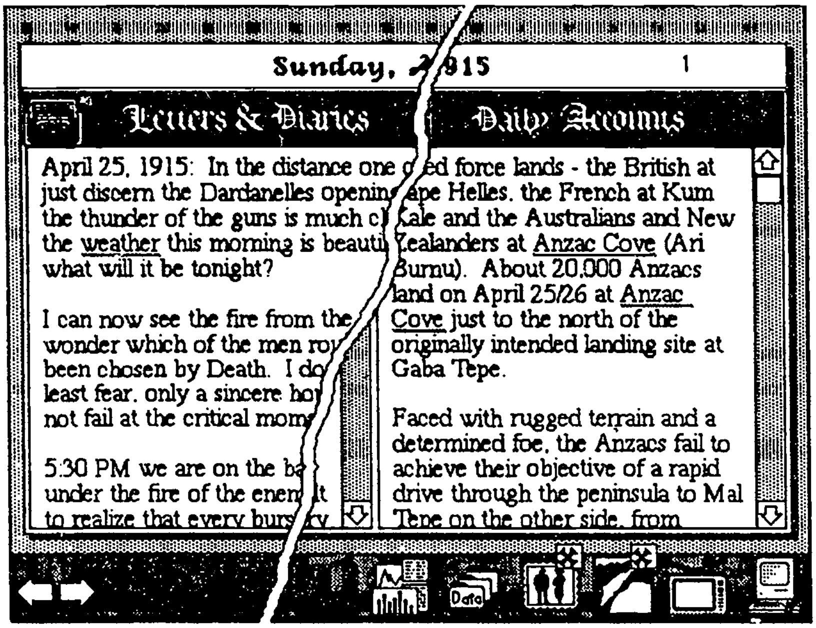


Figure 2. A sample of a typical diary stack screen

The date and days from landing information are also provided to give chronological references. It also includes the day of the week which adds an extra dimension to the questions which may be asked. Since April 25 was the date of the landing, days in early April are negative. These have been included to show that the campaign did not just begin on April 25, a fact which is easily overlooked. Words or phrases which are underlined indicate that clicking on this word or phrase will link to another part of the system.

Navigation is performed using a variety of buttons as well as a unique calendar interface. The Calendar Palette (see figure 1) is used to select particular dates rather than scrolling from one day to the next or using and index or table of contents. It provides an efficient access mechanism, using a metaphor that users readily identify.

Access to the other parts of the system is via a consistent set of icons along the bottom of each screen. Clicking these icons usually takes the user to the first card of that part of the system. Some days have a special reference marker attached to either of the Picture, Map or Film icons. This is used to indicate that there is a specific picture, map or film clip that relates to the events of that day. Clicking on the appropriate icon will take the user directly to that reference material.

The Statistics stack

This contains a miscellaneous set of data divided into eight major sections as follows:

- Facts & Figures provides statistics on the number of soldiers involved and enlistment data during 1915/1916
- Nations Involved is a summary of the number of troops / casualties etc involved in the whole Gallipoli campaign
- Convoys is a list of all the convoys which sailed with Australian troops, the names of all the ships and their embarkation dates and ports.
- Anzac Units provides a breakdown of all the Anzac units that served on Gallipoli.
- Army Organisation describes how the army was organised at that time and the ranks of soldiers.
- Expressions & Abbreviations is a short dictionary of abbreviations and expressions which were in common use at the time. Some of these were unique to the Anzacs as they served to disguise the discussions both verbal in the trenches and in their letters home. The "slang" or vernacular language gives a unique insight into what life must have really been like. For example in reading about "Anzac stew" one might be

forgiven for mistakenly believing that it represented some type of nourishing meal. In reality it demonstrates the typical Australian humour because it was simply one bacon rind boiled in an urn of water!

- Biographics is a short set of biographics and pictures of some of the key players in the Gallipoli campaign. While much of the material is selected from an Australian point of view, the biographics section does include a number of British, Turkish and German generals.
- The ANZAC Book was a book that was produced by soldiers in the last days of the campaign. It consisted of writings, sketches and cartoons from a large number of contributors. It was later published but some of the sketches were omitted for political reasons. This section contains some of these censored cartoons. While they may not seem particularly sarcastic or damaging today, in terms of the political climate in Australia at the time, they were very controversial. In some instances, they have never been published before.

The Pictures stack

The archives of the Australian War memorial contain many black and white photos of scenes related to the Gallipoli campaign. While a good many of these may have been published at various times in the past, the extensive selection provided here serves not only as a pictorial history in its own right, but also a unique insight to the day to day activities of the troops. Many pictures of mundane tasks such as the carrying of water (figure 3) which occupied a great deal of the activities of the soldiers, have been included to reinforce their accounts from the diary. Without the use of the picture compression technology provided by Apple's QuickTime system, such pictures would not have been a viable proposition.

Figure 3. Carrying water in the trenches.

The Films stack

Very little film footage was ever taken at Gallipoli and only a few fragments remain. This stack contains parts of that remaining black and white footage as well as segments from the feature film Gallipoli. Only small sections which are historically accurate recreations of the actual events were used from the feature film. Like the pictures, the film clips provide a reinforcement of the written accounts, in a way which has not been done before. For example, the famous charge at the Nek during August in which 500 soldiers were killed in an area just over the size of a couple of tennis courts is recreated with such stunning effect that could not be portrayed by any other means. The ability to control individual frames from within the quickTime movies also adds to the flexibility of the system with individual frames being the focus of discussion.

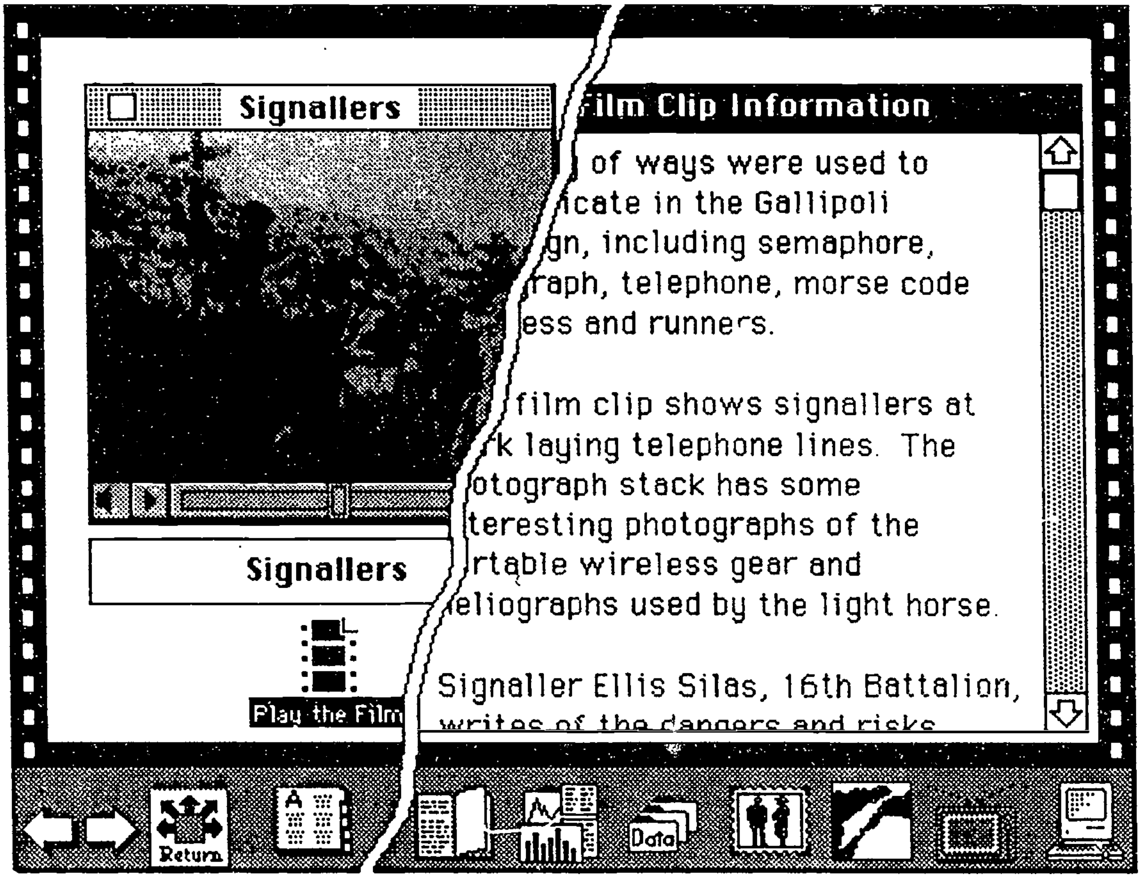


Figure 4. Sample of a typical film stack screen which has been cut to fit within the margins

As in the case of the photographs, the use of QuickTime compression was critical to the success of the result. Film clips which totalled over 60 megabytes were able to be compressed to occupy less than one third of their original size without an appreciable loss in quality.

The Maps stack

Maps are an integral part of the paraphernalia of war. Included in this stack are maps that may be used to locate landmarks which are referred to in written accounts, trench maps showing troop dispositions at various times, maps showing the progress of particular battles or raids and even animated demonstrations of trench digging. The maps play an important part in developing a sense of the scale of the activity, particularly in realising that this all took place in such a small area.

The Database stack

While it may have provided the genesis of the project, the database stack is the least developed of all the components. All of the original records have been converted and an inquiry system provided for searching. Most of these records were contributed by students from all over Australia by doing research on their own local identities who may have served at Gallipoli.

Future Developments

Gallipoli is only a start. Historical information systems will soon be commonplace. As repositories of data such as libraries, archives and museums begin to store copies of their collections in electronic form, the dissemination of the vast amounts of information held in these huge granaries of knowledge will for ever change the face of education. The Australian War Memorial is to be commended for its foresight and generosity in opening up this part of its collection. The technology is poised, as if about to open a door that leads to a scene that one can only guess at.

Cooperation Support in Computer-Aided Authoring and Learning

Max Mühlhäuser and Tom Rüdebusch
University of Karlsruhe, Telecooperation Group
P.O.B. 6980, 76128 Karlsruhe, Germany; [+49] (721) 608-4790

Abstract: This paper discusses the use of Computer Supported Cooperative Work (CSCW) techniques for computer-aided learning (CAL). The work described was started in the context of project Nestor, a large joint effort of german universities and Digital about cooperative multimedia authoring/learning environments (Mühlhäuser and Schaper 1992). We will motivate the specific importance and benefits of CSCW in the CAL domain (chapter 1). Chapter 2 concentrates on a very generic support system, called GROUPIE, which we developed, and shows how it was tailored to CAL; sample cooperative courseware will also be briefly discussed. Chapter 3 covers the second important kind of CSCW support, pre-built cooperative tools, and illustrates their use in a framework for cooperating authors, called DIRECT.

1 Motivation and Classification

The importance of CSCW for the CAL field has been recognized early on, cf. (Wilton 1985; Ward 1991; Smith et al. 1989). As to the authoring side, CAL research in the past years has shown that profound expertise in several disjunct areas is required in order to develop appealing and effective instructional material: ***domain knowledge*** (about the area to be taught), ***instructional design knowledge*** (about instructional strategies, learner and knowledge modeling, courseware lifecycles), and ***media and interface design expertise.*** A single individual can hardly cover all these areas in detail. As a consequence, several people have to cooperate during the process of courseware development in order to obtain reasonable quality. Quite often, such experts work at physically disjunct places. Thus the need for authors to cooperate.

On the learner side, the situation is even more urging: after decades of ITS research, it is recognized that computers can not entirely replace human tutors. As a consequence, learning environments should offer the possibility for the learner to get assistance by knowledgable people (the author, a tutor). Cooperation with authors emphasizes usually on iterative 'feedback loops' that help both the learner (in understanding the subject or the courseware utilization) and the author (in getting hints for improvements). Using more advanced 'liberal' approaches, the distinction between authors and learners even becomes blurred, e.g., if a group of authors tries to acquire knowledge about a domain.

An even more important motivation for cooperation on the learner side is deeply rooted in the modern educational system. The pressure on pupils and students to reach high standards 'produces' rather isolated graduates, accustomed to working alone. Modern media and, not to the least, PCs have largely contributed to this 'lone wolf syndrome'; CAL runs the risk of aggravating this negative trend. At the same time, industry and economy move towards the 'global village'; hardly any reasonable task can be achieved by individuals any more, teamworking skills are strongly required. It is therfore crucial for the success of CAL that courseware addresses cooperation among learners. This goal can only be achieved if teamwork aspects are included in as much courseware material as possible.

To summarize, four major categories of cooperation can be identified for CAL: **author-author, author-learner, tutor-learner, and learner-learner** (note that an individual may play several of these roles in different contexts).

Generic authoring support: interestingly, all but the first one of these categories must be considered in the courseware itself. Since courseware is individually designed for each domain and purpose, pre-built cooperation tools for learners are not very helpful. Early attempts like the provision of electronic mail connections or even videoconferencing have proven to be of limited use. Such approaches require the users to talk *about* the courseware instead of *using* it cooperatively. E.g., for a tutor to provide efficient help, he might need to 'look at' the learner screen, 'ask' the courseware about the learner history, and make remote input to the courseware. Efficient learner cooperation requires that the notion of different learners is 'known' to the courseware and supported. To summarize, generic authoring support for cooperative courseware is crucial in an authoring/learning environment (cf. chapter 2).

Cooperative authoring tools: the first category above, author-author cooperation, is more suited for pre-built cooperative tools than the learner-related scenarios. Chapter 3 will concentrate on this issue in more detail.

2 Generic Authoring Support for Cooperative Software

2.1 Domain-Independent Generic Approach

We investigated many existing cooperative systems in diverse domains, drawing three major conclusions (Rüdebusch 1993). 1: The field of CSCW is lacking a comprehensive and systematic taxonomy; the well-known time-space-categorization (Johansen 1988) is useful but much too restricted. 2: Most existing cooperative applications represent dedicated implementations (wrt. application domain and interaction patterns). 3: All existing CSCW implementations were large programming efforts, again and again realizing similar cooperation functionalities from scratch.

In the GROUPIE (Group Interaction Environment) development which was carried out in the Nestor project, we first established a common model and taxonomy of human-human cooperation in distributed systems. Based on this model, we realized a universal development and runtime support system for cooperative software. Last not least, we used GROUPIE to build example cooperative applications. In GROUPIE, the taxonomy and model of *cooperation* in distributed systems is viewed as an aggregate of two lower-level concepts, namely *interaction* and *coordination*.

A) Interaction: this term denotes any user-initiated action in a CSCW context. It is the basic means of cooperation. Just as actions in a single-user scenario are performed upon objects (e.g. resizing a rectangle in a graphical editor), interactions are performed upon so-called team objects. This new concept extends the well-known object metaphor from single-user work to teamwork. (In the example, resizing the rectangle could be made visible to other group members.) Interactions can have a variety of characteristics, as depicted in fig. 1, "Interaction characteristics".

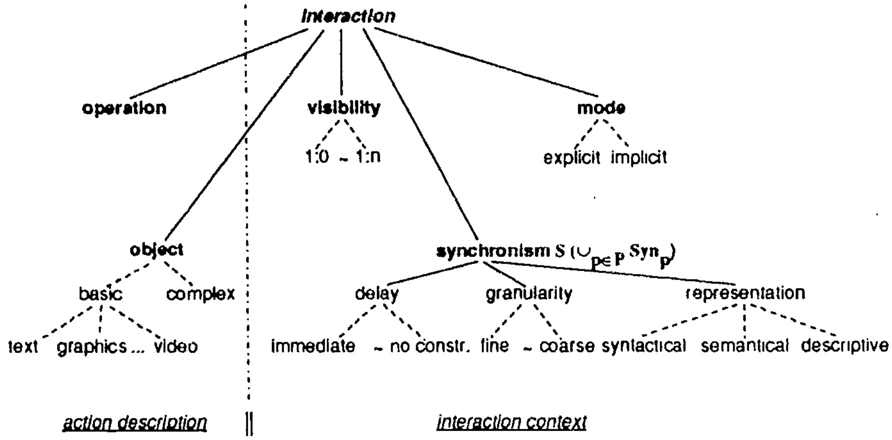


FIGURE 1. Interaction characteristics

In the outset, an actions is defined as usual as an operation performed on a (basic or complex) object. The action description is then augmented to an interaction via the *interaction context*, comprising three components.

The *visibility* component specifies the partners for whom the current interaction is visible (e.g. group members taking part in the cooperative editing of a document). By setting the visibility to 1:0, a single-user action can be specified as a special case of an interaction. (In general, team objects provide a compatible extension of single-user work towards teamwork, as private work is always consistently supported as a special case of group work.)

The *synchronism* component is again subdivided into three components. First, a maximum time delay between the initiation of an interaction and its visibility at a partner can be given. Second, the granularity describes the quantum of sub-actions to be transmitted to a partner (e.g. when resizing a rectangle, each pixel movement, each interval of 10 pixel movements, or only the completed operation can be transmitted). Third, the representation at a partner can be syntactical (exactly the same view), semantical (consistent data but different views, e.g. a pie chart and a bar chart for an integer array), or descriptive (only indicating that an operation has been performed, e.g. 'max has edited figure 1'). Synchronism can be described for each partner individually (e.g. reflecting a tightly cooperating group in a local network with loose connections to a geographically remote member).

The *mode* of an interaction is explicit when a team object is deliberately sent to a partner (e.g. when sending a document via email). Interaction takes place implicitly when acting upon shared team objects (e.g. in a multi-user editor).

B) Coordination: this concept is orthogonal to interaction (i.e. each coordination characteristic can be combined with each interaction characteristic). We speak of basic coordination (or micro coordination) when low-level aspects like authorization and concurrent access are regulated. Complex coordination (or macro coordination) recognizes composite tasks and regulates work flows or conversation structures. In our model, a coordinating framework can be specified (in the beginning independently) for teams and for tasks. In a second step, a team and a task description are combined into a cooperation description. This fosters reusability to a high extent.

Coordination comprises team-, task-, and cooperation-specific aspects. A ***team*** is described by team-roles and an interaction structure. Team roles abstract from specific users in the team context (e.g. leader, member, protocol-keeper). Interaction structures define interaction paths between team roles (fully democratic, strictly hierarchic, etc.).

A *task* is more complex to describe. Task roles abstract from specific users again, here however as related to the task (e.g. course author, tutor, student). Team objects are the work subject within the task. For each team object, authorization of different task roles and conflict resolution for concurrent access can be specified. Complex tasks can be decomposed into sub-tasks recursively. Constraints watch over sequences of sub-tasks, temporal and logical conditions.

A *cooperation* associates a team with a common task by mapping team-relevant and task-relevant roles.

Coordination takes place on an interaction granularity, i.e. each interaction during task performance is checked against the coordination descriptions. Coordination can also influence task performance actively by guiding team members (e.g. suggesting a next sub-task to work on, based on the coordination descriptions).

Development support in the GROUPIE system comprises a library of team objects, formal languages (for team, task, and cooperation description) and cooperative tools and methods. Thus, CSCW functionality is implemented on an object granularity, as opposed to the common approach of tool granularity. Teams and tasks are not strictly predefined but can dynamically be adapted to changing needs. A cooperative method for developing CSCW systems in teams has been implemented and can be reused or further refined (bootstrap approach).

The runtime system: GROUPIE *distribution support* provides a high-level object-oriented interface based on (home-brewn) Distributed Smalltalk. *Distribution management* decides about migration, replication, and consistency. Cooperative applications are embraced by (guiding and checking) *coordination support, interaction handling* interprets and realizes an interaction context. It is possible to integrate existing single-user applications in a cooperation-transparent way (for the term cooperation-transparent, cf. chapter 3).

2.2 CAL-Specific Extensions to GroupIE and Sample Courseware

According to our concept of generic support for cooperative software as introduced above, adaptation to specific application domains such as CAL is technically simple. Existing base classes need to be adequately refined. Much more difficult is the question of "what are adequate team objects and strategies for cooperative courseware?".

Cooperative problem solving can, in a first approach, be supported by simply providing an information space to the learners that is freely accessible. Information about the learning domain can be retrieved both cooperatively and individually if all information objects are realized as team objects (e.g. documents, video sequences). *Student evaluation* requires dedicated learning objects (e.g. questionnaire, multiple choice documents for cooperative completion); tutors (with specific privileges) may be involved. *Cooperative strategies* are even more interesting: well-known learning strategies for individuals (drill&practice, tutorial, exploratory learning, etc.) must be complemented by cooperative ones which, e.g., take into account different roles of learners. E.g., procedural knowledge of legislation in the European Community may be taught based on various roles of delegates in the Commission; a cooperative strategy for information acquisition is described in ch. 3; a motivational approach might use the paradigm of 'multi-user dungeons', etc.

coopEC (cooperating within the European Community learning domain) is as sample courseware for validation of our concepts. coopEC teaches about the EC and features a cooperative problem solving process and working on various documents. Four basic team objects specific to CAL have been developed in the first version: a questionnaire, a multiple choice document, a map of Europe with hot spots, and a free-form text/graphics document. It supports dynamic transition between private and group learning, flexible coupling modes, and both on-line conferencing and asynchronous document exchange. Once enrolled,students can also be sent documents while logged off.

Fig. 2 shows the main window of coopEC. The sub-window at the top is used for cooperation control. The very detailed set of possible interaction characteristics as introduced in chapter 2.1 has been customized into a few user-selectable combinations, each accessible via one respective button, that seemed most appropriate for this specific setting.

The user can, at any time, choose 'private' work or cooperate with his partners. To cooperate, he can 'share' documents (with the whole group or a sub-group) or 'send' a document to any number of partners. The closeness of a cooperation can be determined by selecting one of three 'coupling' buttons. With 'tight' coupling in share mode, all partners have the same view on the shared document. Any operation, like typing a single character, is visible for each of the partners. Full telepointing is supported, as the movement of each partners' mouse pointer is overlaid onto the shared document. In send mode, tight coupling results in a mail window automatically being brought on top of all other windows for each receiver. 'Medium' coupling in share mode transmits only completed interactions. E.g. when filling out the questionnaire, only complete answers are transmitted to the partners. 'Loose' coupling, whether in share or send mode, just prints a textual information about the operation performed in the message area at the very bottom of the main window. The sub-window below cooperation control provides meta information about users and modes. coopEC supports two roles: tutor and learner (indicated in brackets). Users listed as inactive have never accessed the course. A unique color is mapped to each user. This is extremely helpful when sharing a document (to distinguish multi-user input and telepointers). Green is reserved for the tutor, making him easy to identify.

Four buttons 'questionnaire', 'multiple choice', 'euro map', and 'note' are provided to select a document to work on (details skipped for the sake of brevity). As an example scenario, a learner could decide to start a cooperative questionnaire by answering the questions he already knows. He may then invite some of the enrolled students to share this partly completed questionnaire.The sub-group can then cooperate by inserting or correcting answers (the color of each answer text dynamically changes to the unique color of the current editor). Telepointers can be used to indicate issues currently in discussion. The learners may decide to involve the tutor and finally submit the questionnaire.

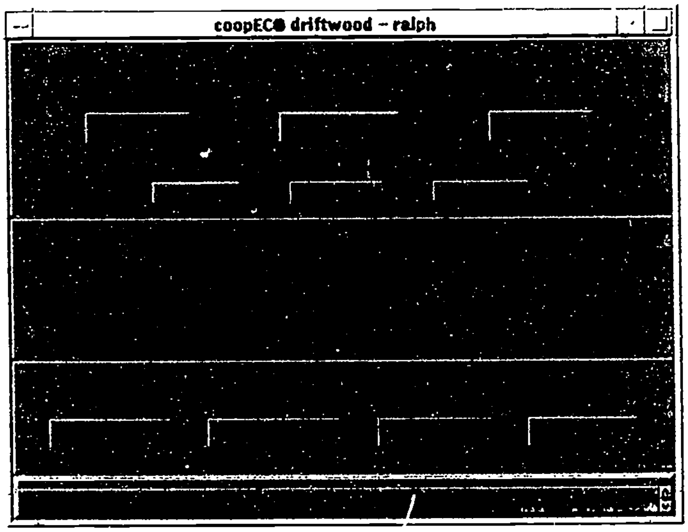


FIGURE 2. Top-level user interface for coopEC

3 Cooperative Authoring Tools and Lifecycle

Commercial courseware development has in the past been based on rather prescriptive lifecycles. New management theories and cooperation trends suggest more liberal approaches, as does the academic working style. We will concentrate on such a more liberal approach in the remainder (more prescriptive lifecycles can easily be supported). The resulting authoring system Nestor-ADP is comprised of cooperation-transparent tools, cooperation-aware tools, and the cooperation-aware lifecylce support environment DIRECT.

Cooperation transparent tools: this term refers to tools which have been built with a single user in mind. The idea is to extend them transparently for use from multiple user interfaces, possibly in a distributed system (the advantage that existing tools can be augmented is paid with limited cooperation functionality).

Nestor tool "shX": In the context of our project Nestor, generic cooperation support for cooperation-transparent software has been developed, called "shX". shX allows prebuilt Xwindow™ -based application software to be used from multiple workstations (even without recompilation or rebinding). All output from the application is replicated to all participating workstations. shX offers different so-called floor passing schemes, ranging from 'prescriptive token-based' (only one user has control at a time) to 'anarchy mode' (uncontrolled multiple input). New schemes may be added, such as role-specific ones (e.g., 'tutor-learner', where the tutor always has superior input rights).

IDE-shX: since Nestor tries to support the entire courseware lifecycle, we built a specific tool for the early courseware design phase, called 'instructional design editor'. This tool allows to arrange instructional goals and objectives graphically into a semantic network, relating these instructional issues to domain knowledge. IDE 'understands' several instructional strategies. IDE has been extended by shX in order to allow cooperative courseware design.

CSE-shX: IDE-based designs can be 'compiled' with one of the instructional strategies implemented in IDE, generating template course structures for the 'course structure editor' CSE. CSE is based on a high-level graphical notation for the flow of control and actions in the courseware, called 'instructional transactions. The template generated (by IDE) for CSE provides the courseware skeleton which has to be complemented by concrete screen and media designs and by fine-tuning for the instructional transactions. A cooperative shX-based extension of CSE has been built. In the long run, we plan to implement cooperation-aware successor tools to IDE and CSE based on GROUPIE.

Cooperation-aware tools: as mentioned above, cooperation-transparent applications are very restricted. E.g, they can hardly be used if teams work together at different times. Since the courseware lifecycle is a long-living activity, such asynchronous cooperation is however crucial. Therefore, we had to build several cooperation-aware tools plus a cooperation-aware lifecycle framework encompassing the (cooperation-transparent and -aware) tools. In addition, integration of audio and video conferencing support was found to be an absolute requirement for efficient cooperation of physically distributed authoring teams.

GROUPIE itself is, for the moment, our most important cooperation-aware environment. In addition, the following cooperation-aware project management tools were built: a meeting coordination tool which interworks with private calendar tools and with the other tools listed here; a group mail system (which uses the user's standard mail for sending/receiving mail); individual calendar tools; a time / task planner based on critical path networks CPN. Apart from that, tools for audio and video conferencing have been implemented, featuring dynamic conference management. The audio conferencing is based on our low-cost SCSI^{TM}-based hardware audio extension for workstations. The video conferencing support uses vendor-supplied frame grabbing boards and a home-brewn software video codec, called SMP (Neidecker-Lutz and Ulicheny 1993).

Cooperation-aware lifecycle support: at a first glance, one may decide to base a cooperative framework for courseware lifecycle integration on traditional courseware development processes. However, it was already argued in chapter 2 that modern views of cooperative work tend to seek new, more liberal management and organization approaches than the model of subsequent "phases" (like 'define', 'design', 'develop', 'deliver',...). In fact, if courseware development should lead to new interesting solutions, the outcome and goals are not very clear at the outset: courseware development becomes an 'ill-structured problem'. An excellent approach to coping with such ill-structured problems has been presented in (Potts 1989), called 'issue-based design'. In our project, we adapted this concept to the problem space of courseware development and implemented a cooperative framework accordingly. The framework was called 'DIRECT' (distributed research and engineering for cooperating CAL teams).

The courseware-adapted issue-based design approach shall be briefly described here. According to this approach, the problems associated with developing a specific courseware are structured into 'issues' (each issue can be hierarchically separated into sub-issues). For each issue (or sub-issue), different approaches may be identified over time (these approaches are called 'positions' according to the original method), and arguments supporting or objecting to an approach may be collected in the cooperating group of authors. In DIRECT, the encompassing graphical user interface supports the cooperative construction of complex, hierarchically structured issue-position-argument graphs. A particular value of the system stems from the provision of authoring-specific issue-position-argument templates which are predefined for reuse.

According to the method, the authoring team has to opt for one of the alternative approaches to an issue. At this point, a number of possible so-called 'steps' are offered to the authors which they can choose from in order to resolve the issue (an example for a standard step is the development of a courseware module). It is (mainly) here that the above-cited tools are 'hooked' into the encompassing framework: if a specific step is chosen, a kind of 'workflow support system' is started which manages the coordinated use of the above-mentioned courseware design/development tools, project management tools and audio/video conferencing tools. Our experience has shown that this integral approach has a lot of advantages over the uncoordinated use of individual cooperative tools. As an example, if a step starts with the definition of a task force (e.g., assigned for the development of a specific courseware module), then subsequent tools can be automatically provided with the names and network addresses of the relevant group participants.

The screenshot in fig. 3 shows the screen of a DIRECT user, a partial display of an issue-position-argument graph (upper right) and some cooperative tools which are actually in use by the author (e.g., IDE-shX at the lower left).

4 Conclusions

We have shown that two basic areas of cooperation support have to be distinguished, author-author cooperation and learner-side cooperation. Learner-side cooperation must be supported by generic development support for cooperative courseware rather than by tools. Author-author cooperation involves multiple cooperation-aware (and maybe cooperation-transparent) tools and can be best accomplished with the help of an encompassing framework. The latter was described based on a liberal courseware lifecycle. Stable versions of GROUPIE, coopEC, cooperative authoring tools and DIRECT as described are actually running on our premises and at several external sites.

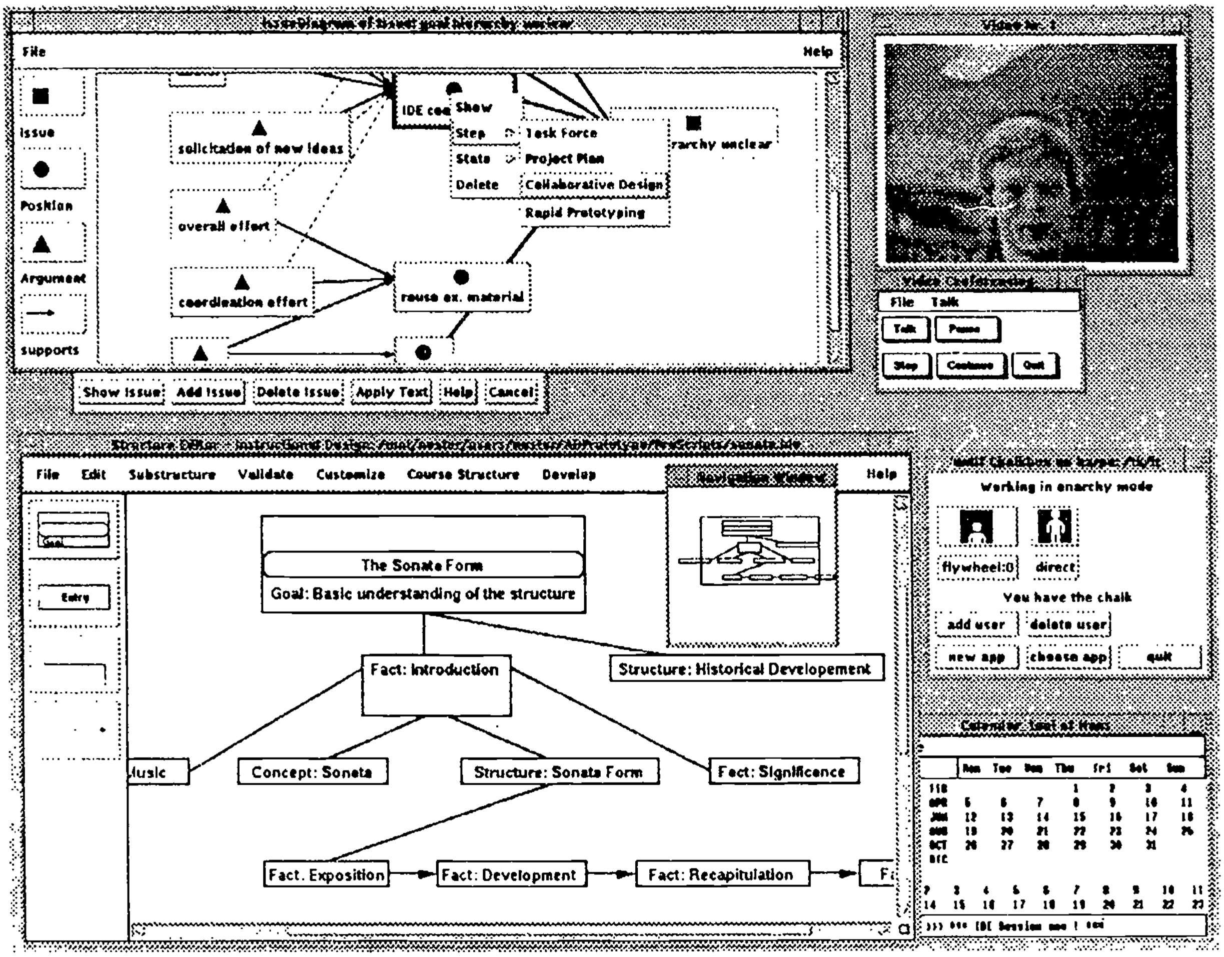


FIGURE 3. Sample Screenshot of the DIRECT Courseware Lifecycle Framework

5 References

Johansen, R. (1988): Groupware: Computer Support for Business Teams. *The Free Press, New York.*

Mühlhäuser, M., Schaper, J. (1992): Project NESTOR: New Approaches to Cooperative Multimedia Authoring/ Learning. *Tomek, I. : Computer Assisted Learning, Springer Berlin, 453-465.*

Neidecker-Lutz, B., Ulicheny, R. (1993): Software Motion Pictures. *Digital Technical Jl. 5. 19-27.*

Potts, C. (1989): A generic model for representing design methods. *in: Druffel, L. (Ed.): Proc. 11th Intl. Conf. on Software Engineering, Pittsburgh, PA. ACM Press, New York, 217-226.*

Rüdebusch, T. (1993): CSCW - Generic Support for Teamwork in Dist. Systems. *DUV Wiesbaden,* in German.

Smith, R B. et al. (1989): Preliminary Experiments with a Distributed, Multi-Media Problem Solving Environment. *in. Proc. ECSCW 1st Europ. Conf. on Comp.Supported Coop. Work, 19-34*

Ward, D.R. (1991): Boosting Connectivity in a Student Generated Collaborative Database. *in: Proc. ECSCW 2nd Europ.Conf on Comp Supported Coop.Work, Kluwer, Amsterdam, 191-201.*

Wilton, J.A. (1985): Structuring Learning Networks. *in. WCCE, World Conf. on Computers in Education, 491-496.*

Rapid Prototyping of Computer-Based Presentations using NEAT, Version 1.1

Tomasz Müldner
Jodrey School of Computer Science
Acadia University, Wolfville, Nova Scotia, B0P 1X0, Canada
email: tomasz.muldner@acadiau.ca

Abstract: NEAT, which stands for iNtegrated Environment for Authoring in ToolBook provides templates and various facilities for the rapid prototyping of computer-based presentations. Outlines of the presentations can be created and modified. NEAT supports reusability by allowing the author to store graphics objects and entire pages in a repository, and to retrieve them when needed. Finally, NEAT provides support for communication between members of the authoring team and between the team and the learners.

1 Introduction

Authoring systems have been used for many years and a number of interesting applications (courseware) have been developed, see [MAU90, ALE91]. However, currently existing authoring systems seem to be quite weak in the support for the following areas:

- availability of tools for rapid prototyping
- reusability of concepts and objects, such as graphic objects
- hypertext and hypermedia facilities
- support for building meta-tools, such as icons to execute macros
- extendibility.

While the author is provided with considerable support at the microscopic level, for example specialized editors that are used to create a single frame, text, graphics or animation to appear in this frame, there is little support at a macroscopic level. Thus, it is not possible to prototype courseware and leave details for future development. Another important deficiency of the current authoring systems is that the authoring team has to make an early decision as to what the level of expertise of the expected audience is. For example, there may be beginning learners who do not have any knowledge of the subject, and other learners who have some, or advanced knowledge. The advanced learner would find it boring to go over the entire material necessary for the beginning learner, but may require additional information that is not made available to the beginner. Thus, it would be useful to provide different views of the same material so that each learner can find a view appropriate for his or her background.

Most, if not all, existing authoring systems do not separate the creation and maintenance of the knowledge base, and the presentation of this knowledge (for the description of the second generation of authoring systems, see [MER89]). Members of an authoring team represent different expertise and abilities to work on the development of computer based presentations. Very often there is no single author who creates courseware. Instead, there is an authoring team, whose members represent various types of expertise. For example, one member of the team may be a subject-area expert, with an ultimate knowledge of the subject on which courseware is to be developed. This expert may have limited knowledge or time to actually develop courseware. Other members of the team may be designers, specialists in learning strategies, and programmers;

for example, programmers specializing in graphics, animations, or multimedia. Only subject-area experts know what should be included in courseware, and so they should prototype this courseware. This prototype can be further refined and modified by other members of the authoring team, provided that there is a simple and effective way of communicating ideas between various members of the team.

NEAT is a specialized authoring system designed to alleviate some of the problems described above. Version 1.0 provided limited support for views and question templates, see [MAY93]. Here, we describe NEAT 1.1, designed and implemented by the author of this paper in the summer of 1993. NEAT can be used by authors who have a very limited knowledge of computer systems and no programming experience. These authors can communicate with other members of the authoring team using annotations, hypertext links and highlighting. Computer-based presentation developed with NEAT is called *neatware*. NEAT is a ToolBook application (ToolBook is produced by Asymetrix), and neatware, produced with NEAT is an application that can be executed with the ToolBook run-time system.

The rest of this paper is organized as follows. First, in Section 2 we describe neatware. Then, in Section 3 we describe NEAT, and in Section 4 prototyping with NEAT. Section 5 briefly describes the implementation of NEAT, and the Conclusion presents several examples of neatware.

2 NEATWARE

Neatware is a specific type of courseware, based on a book metaphor. However, neatware can consist of *multiple views*, for example a *beginner* view and an *expert* view. One can think of these views as separate books, and the user can switch between various views. As in a traditional book, each view consists of *chapters*, a *table of contents*, and an *index*. Each chapter consists of sections and pages; a section consists of pages and sections.

As in the real world, the appearance of these books can be altered in several ways by the reader during the learning process. Notes can be kept separate from the book, traditionally in a notebook, known as *global notes*. Notes can be also kept on each page being read, traditionally in the margin of the page, and therefore known as *margin notes*. *Bookmarks* may be placed in various pages of the book. Words on each page can be *highlighted*, in one of several colors.

Unlike in traditional books, an electronic book has other worthwhile features. *Example* pages and *windows* can be accessed from various pages, possibly with a choice of the user's *preference* of example type (e.g. Pascal, C or Modula-2 example), providing more insight into specific details of the neatware. As each page is read, its name is stored in a list known as the *history*, which the reader can use to return to recently visited pages. *Hypertext links* can be used to move around in the book, moving the user to information which may be more relevant. Also, the table of contents contains *bread crumbs*, or *footprints*, which show which pages of the book have already been read. Of course, neatware also includes various *navigation tools* to allow the reader to change pages and access all of the above features.

In summary, each *neatware* has the following characteristics:

- multiple views of the same material; each view consists of chapters; chapters consist of pages
- electronic index which can be modified by the user
- bread crumbs (or footprints), showing the progress of the student
- margin notes that appear on each page
- global notes that resemble a sheet of paper attached to the book
- electronic bookmarks which can be used to save references to the selected pages
- history, showing the list of pages most recently visited by the user
- hypertext links which can be used to read the material in a non-linear fashion
- electronic highlighting of the selected text
- examples which can appear on every page. Each example is associated with the button and can be activated by the learner by clicking on this button. There are three kinds of examples: an example which appears on a separate page; an example which appears in a window on the same page; and an example which can have several appearances, depending on preference that can be selected by the learner (the user's preference may be based on his or her knowledge when studying the book, for example, on knowledge of Pascal or FORTRAN when studying C)
- a repository of examples with a hierarchical structure. These examples can be modified by the user. A sample neatware page is shown in Figure 2.1.

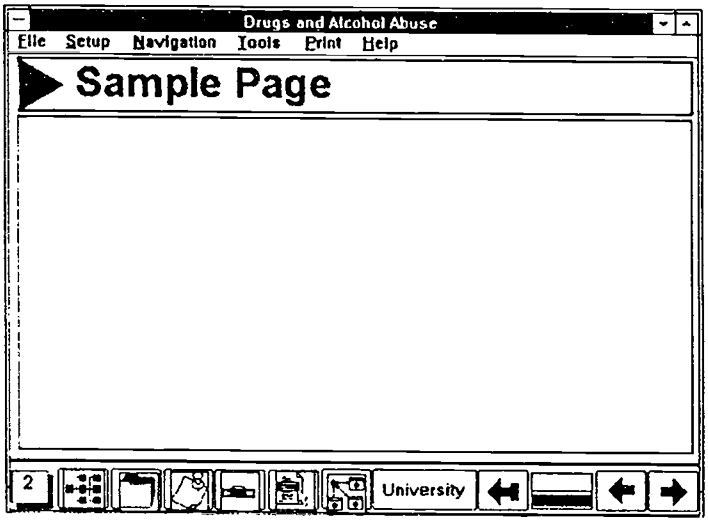


Fig. 2.1 A sample page.

At the bottom of the page, there are various icons, used to navigate through neatware, using one of many navigational features and tools provided by NEAT. In Fig. 2.1 icons represent respectively (from left to right) the page number icon, the table of contents icon, the index icon, the global notes icon, the local notes icon, the highlight icon, the hypertext icon, the view icon, the RETURN icon, the slider icon, and the previous and the next page icons. Menus are provided at the top of the page. Menus are used to perform actions such as "bookmark a page", or "save neatware".

In the rest of this section, we describe two tools used for the communication between the various members of the authoring team and between authors and learners, that is hypertext links and annotations. For more details of neatware, see [MUL93a].

2.1 Hypertext Links

Hypertext means a non-linear or associative structure of pieces of text. Pages in neatware may have text which includes the so called *hypertext links*, which are mouse-sensitive text phrases. When such a link is clicked with the mouse, the user is moved to another piece of text, or another page. To help the user find hypertext links, they are underlined. There are three kinds of hypertext links:

- *help* hotword, underlined in blue; produces a window on the same screen
- *forward reference*, underlined in red; move the reader forwards
- *backward reference*, underlined in green; move the reader backwards.

Hypertext links can be created and modified during the development of neatware. Thus, some of them can be used for the communication between members of the authoring team, while others can be left for the learners. Neatware hypertext links can also be created and modified by learners.

2.2 Annotations

Annotations are used by the user (both the author and the learner) to highlight important, or difficult sections of courseware. Neatware provides the following types of annotations:

- *highlighting*. A part of the text on the current page can be selected and highlighted by coloring it with one of the available colors. Unlike conventional highlighting, this highlighting can be removed
- *margin notes*, similar to those made on the margin of the page. In version 1.1, margin notes can be stored in both a text and a sound form
- *bookmarks*, which are labels that can be associated with the selected pages. These labels are texts defined at the time the bookmark is created. The user can use existing bookmarks to move to the corresponding pages, add new bookmarks and remove the existing bookmarks. A special bookmark is used when the user quits neatware. This bookmarks stores the current time and date and it can be used to return to the point from which the book was exited
- *bookmarked notes*, which combine bookmarks with margin notes
- *global notes*, similar to those made on separate sheets of paper, and attached to the book.

The next section briefly describes main features of NEAT.

3 NEAT

The basic principles of the design of NEAT are:

- maintenance of the book structure (that is the structure of chapters, sections, etc.) is transparent to the user. Thus, the user can insert, delete, copy and move chapters, sections, and pages without having to modify any of the navigation tools, such as "go to the next page"
- automatic creation of computerized drills using *templates* for six types of questions; multiple choice, fill in the blanks, numeric analysis, position analysis, matching and textual analysis
- support for *reusability* by allowing the user to store objects and pages for future use
- the development tools do not show up in neatware, so that the user working with NEAT has a clear idea as to what the final product will look like. For this reason, NEAT consists of a series of menus that are used by the user in order to create pages, objects on these pages, etc.
- the user has to switch to the ToolBook's author mode as rarely as possible
- from NEAT, the user can inspect scripts and properties of neatware objects.

There are three tools to support reusability:

- *graphics library* stores graphic objects. The library browser provides two interfaces to the set of objects stored in the library; textual and visual interfaces. The textual interface is in the form of a list of names of all existing objects. The visual interface allows the user to traverse the list of all objects, stored in iconized forms
- *shelves* store pages. For each type of a page, there is a separate shelf. When the user removes a page, it can be stored on a shelf, rather than completely removed. The user can copy or move pages from a shelf into neatware. Shelves are limited to a single book, that is the user can not copy pages from a shelf stored in one neatware into another neatware
- *desks* store pages. Desks are similar to shelves, but they can be used for moving pages between different neatware.

The user can work on at most one neatware at any given time, but it is possible to switch between various neatware. Thus, the user can leave neatware in a certain stage of development, and continue this development at a later time. More details of the NEAT environment are described in [MUL93b]; the next section concentrates on prototyping.

4 Prototyping with NEAT

The user can start developing neatware by creating the outline consisting of chapters, sections and pages. Each of these pages may be initially blank or contains only text, and its details can be decided later on. In order to create and maintain the structure of neatware, the user has access to the following operations:

- *go* to the selected page
- *insert* a new page. Here, the user selects the desired type of a page, for example a chapter, a section, or a preview page. The user also selects, in the table of contents, the page *after* which the new page is to be inserted
- *rename* an existing page
- *delete* an existing page. Entire sections, chapters or views can be deleted. The user has an option of storing pages being removed on a shelf, or completely removing them
- *copy* an existing page. Entire sections, or chapters can be copied within the same view, or between different views
- *move* an existing page. Entire sections, or chapters can be moved within the same view, or between different views.

The above operations can be performed through the so-called *Control Panel* page. The author uses this panel to create and modify the structure of neatware. Control Panel is not visible to the learner, who can view the *Table of Contents* page. Below, we describe both types of pages.

4.1 Control Panel

The control panel consists of two parts. The field on the left-hand side shows the list of all pages which are currently present in neatware, see Figure 4.1.

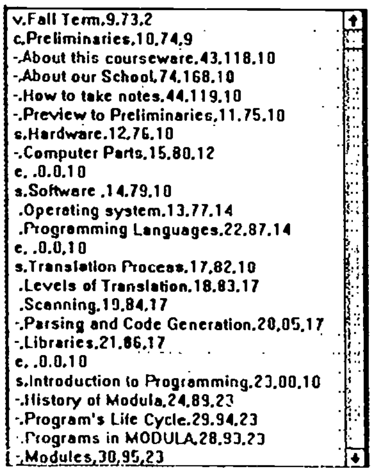


Fig. 4.1 Part of Control Panel

Each line in the above field contains a list. The first character in each line shows the type of the corresponding page:

v	view page
c	chapter page
s	section page
-	any other type of a page.

Additionally, the letter "e" indicates the end of the section. The second item on the list on each line is the name of the page. (The remaining items are not relevant for this paper.)
The second half of the control panel, placed on the right-hand side is a set of buttons to perform various operations, see Figure 4.2.

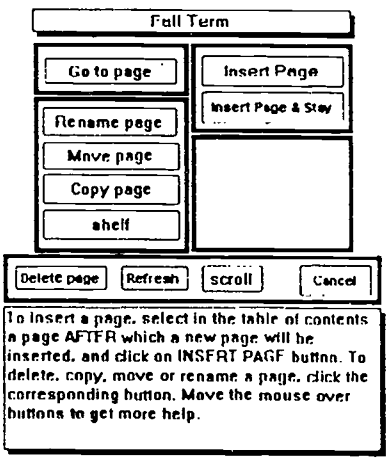


Fig. 4.2 Another part of Control Panel.

The top button, labeled FALL TERM in the above example, produces a pop-up menu with the names of all existing views. This button can be used to change the current view and show its contents in the field on the left-hand side. For most operations, the user selects the page from the list of all pages. For example, when the user wants to insert a new page, he or she has to specify where the page is to be inserted, by selecting the page *after* which the new page will be included. After selecting the existing page, the user clicks the INSERT PAGE button, and NEAT presents the buttons and fields shown in Figure 4.3.

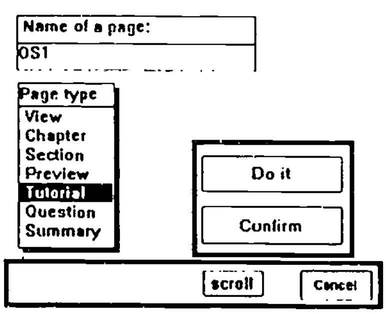


Fig. 4.3 Inserting a new page.

The top field is used to type the name of a new page (OS1 in the above example). The user has also to specify the type of new page by selecting one item in the PAGE TYPE box; by default it is a *tutorial* page. To complete the insertion, the user can either hit RETURN when the text pointer is in the NAME OF A PAGE box, or click on the DO IT button. The user can cancel the operation by clicking the CANCEL button. Clicking the CONFIRM button will allow the user to confirm whether or not the operation is to be performed.

When the copy, or move button is pressed, the user has to select the destination view, that is the view into which the selected page, or pages will be copied or moved (even if the destination page is the same as the current page); see for example Figure 4.4. After the user selected the destination view, the left-hand side of the control panel is split into two parts. The upper part of the screen is for the source view

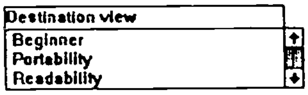


Fig. 4.4 Copying a page.

and the lower part is for the destination view. The user selects the source page or chapter from the upper part of the screen, and then selects a destination page in the lower part of the screen. For example, to copy chapter c1 from view v1 to the beginning of view v2, chapter c1 will be selected on the source screen, and view v2 will be selected on the destination screen.

4.2 Table of Contents

A table of contents contains three windows, representing respectively the list of chapters, the list of sections in the selected chapter and the list of pages in the selected section; see Figure 4.5.

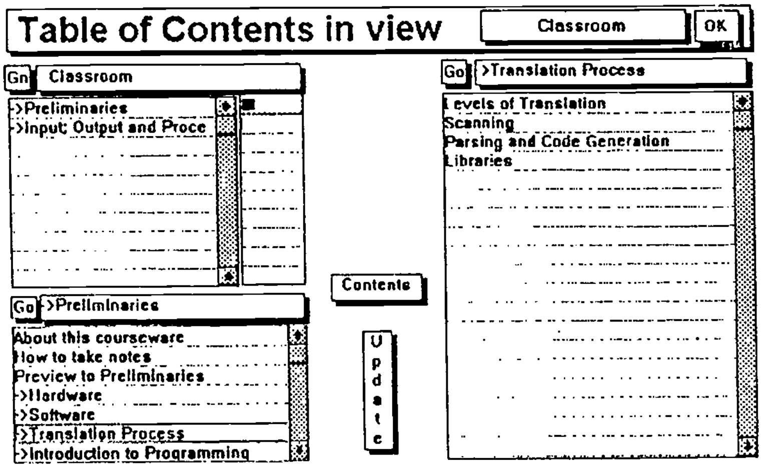


Fig. 4.5 Table of contents.

The user unfolds a chapter or a section by double-clicking it. For each window, there is a title bar, which is also used to unfold the list of pages in this window. For each window, there is also a GO button which is used to move to the selected page. The button placed at the top, right corner labeled OK, moves the user back to the page where the table of contents icon was pressed. The button to the left of this button produces the pop-up menus consisting of the names of all views. The table of contents can be updated by the author after any operation that modifies the structure of neatware; for example after inserting a new page

5 Implementation

We provide only a description of the basic data structures to support prototyping. For more details on the implementation of NEAT, in particular the implementation of hypertext, see [MUL93b]; for the description of OpenScript programming language, see [TOO91]. The implementation of NEAT is based on a tree data structure. There are two types of nodes in this tree:

- definition node, which contains a name but does not have any reference to ToolBook pages
- page node, which contains the name and idNumber of a ToolBook page.

The root of the tree is labeled NEAT (and we will refer to this tree as a NEAT tree); each child of the root represents a single view. A subtree rooted at the view's node represents a structure of this view, that is its chapters, sections, etc. Then, to copy a chapter from one view to another view, we perform a (deep) copy of the subtree rooted at the source chapter.

The NEAT tree is implemented in a single text field, using the so-called array implementation of a tree. The line number in the field is referred to as the *node number* for the node stored in this line. A line in the text field allocated for the node N is of the following form:

node number of the father, N (name of the node), list of children where the list of children consists of pairs of the form (name, node number) or (name, idNumber) depending on whether the node represents a definition or a ToolBook page.

The list of available lines in the field is maintained using two properties, representing respectively the maximum number of lines in the field, and the free list, that is the list of lines that have been deallocated. The maximum number of lines in the field is initialized to a certain value, and is extended whenever necessary. This implementation effectively simulates an "open-ended" array. A single most important concept for all navigation operations is that of a node number. Having a node number available, we can find all necessary information such as the page name, its father, or the list of children. Each neatware has a property whose value is the current node number in the NEAT tree. Moving to another page requires not only the execution a ToolBook instruction to go to this page, but also updating the node number. The basic operation used for navigation is a single step in the depth-first traversal of the tree. This is how the user moves to the next page. Moving to the previous page requires a single step in the *reverse* depth-first traversal.

Conclusion

Our initial experience with NEAT has been very encouraging. During the last tear, the author of this paper was involved in the development of several neatware. The most complete applications are:

- SLADER, neatware on drug and alcohol abuse, see [MUL93c]
- C INTERACTIVE, neatware on teaching programming in C
- MC, neatware on teaching introductory programming in Modula II, used for teaching first year students of Computer Science at Acadia University in 1993/94.

Other neatware are being designed.

During the development of SLADER, we have prototyped large portions of the material, and then presented it to the subject-area experts (specialists on drug and alcohol abuse). We have often used tools such as margin notes, global notes, and hypertext to compile comments provided by these experts and to produce the next version. We have also often modified the structure of neatware; for example the contents of the first version of SLADER has been rearranged as a result of the review conducted by experts from the Counseling Center. Without the support provided by NEAT, we would not be able to complete our work in the same time period.

Acknowledgments

NEAT was originally designed in the summer of 1992 by Tomasz Müldner from Acadia University, Canada, and Stefan Mayer and Claus Unger from Hagen University, Germany. Version 1.0 of NEAT was implemented by T. Müldner in the fall of 1992, and Version 1.1 was implemented by the same author in the summer of 1993. Several students of Acadia University helped at the various stages of development of NEAT, in particular R. Blondon, G. Poulen, Mark Rhodenizer and B. Santosa. C. van Veen implemented most of SLADER neatware.

References

[MAU90] Maurer, H., Tomek, I. *Hyper-G - A Survey*. Report 284, IIG, Graz University of Technology, 1990.

[MER89] Merill, D., Li, Z. and Jones M. *Limitations of First Generation Instructional Design*. Educational Technology, Jan. 1990.

[ALE91] Alessi, S. and Trollip S. *Computer-Based Instruction*. Second Edition. Prentice-Hall, 1991.

[MAY93] Mayer, S., Müldner, T. and Unger, C. *NEAT: An iNtegrated Authoring Environment based upon ToolBook*. EDMEDIA'93, Orlando, Florida, June 1993.

[MUL93a] Müldner, T. *NEATWARE Reference Manual*. Technical Report, Acadia University, Sept. 1993.

[MUL93b] Müldner, T. *NEAT Reference Manual*. Technical Report, Acadia University, Sept. 1993.

[MUL93c] Müldner, T., Duncan, P., van Veen C. *Hypermedia Presentation on Drugs and Alcohol Abuse*, Society for Applied Learning Technology, Multimedia'94, Orlando, Florida, Feb. 1994.

[TOO91] *Using ToolBook*. Asymetrix Corporation, 1991.

Towards automated training of legal problem solving

Antoinette J. Muntjewerff
Department of ComputerScience and Law
Faculty of Law, University of Amsterdam
Kloveniersburgwal 72, 1012 CZ Amsterdam, The Netherlands
e-mail antoi@lri.jur.uva.nl

Abstract: The motivation to study legal problem solving originates from observations in Dutch legal education. Consultation of Dutch research on legal case solving confirmed the assumption that, in using cases in legal education, a method, or explicit strategy, is lacking. However the methods proposed in the research by Crombag, de Wijkerslooth, & Cohen (1977) and Abas & Broekers-Knol (1985) contain some important shortcomings. Research on artificial legal problem solving as proposed and performed by my department is the starting point for the study of automated training of legal problem solving. The domain under study is administrative procedural law. To be able to train students in legal problem solving a thorough analysis should be made of the task involved and the knowledge necessary for accomplishing the task. Because of the fact that training is involved a teachable model has to be constructed.

General description and motivation of the research

If you ask experienced practitioners how they learned to tackle legal problems and to find the appropriate set of norms, the chances are that they will say 'trial and error', 'hit and miss' or similar vague expressions. It is possible that they never actually 'learned' it at all, in the sense of being taught. Few law students get systematic instruction or testing in the technique of legal problem solving. This observation was the starting point of research done in the Netherlands by Crombag, de Wijkerslooth, & van Tuyll van Serooskerken (1972) and Crombag, de Wijkerslooth, & Cohen (1977). They were confronted with the situation that there was no explicit strategy or method for teaching students legal problem solving. The problem solving process of legal practitioners (in this case judges in the field of civil law) was studied. Thinking aloud protocols were gathered and a rational reconstruction of the problem solving process was performed. The idea behind this approach was that the constructed strategy could be used for instructing students on how to solve legal problems. The constructed strategy consists of a series of steps to be taken by the student in the course of solving a legal problem. The strategy was adapted by Abas & Broekers-Knol (1985) and reconstructed into a smaller series of steps. These methods were constructed for use in legal education, not for implementation in a Computer Assisted Instructional (CAI) program or an Intelligent Tutoring System (ITS). This research and my own experiences in legal education lead to the confirmation that legal education is in need of a way to instruct students explicitly in how to solve a case and to provide an opportunity for students to get skilled in solving legal cases. Because there is no time in the official curriculum for training students in this skill, an automated tool could be of use to train students in legal problem solving. However, the problem with the developed problem solving strategies is that they only consist of a series of steps to take by the problem solver (in this case the student). For a method or strategy to be teachable in an artificial problem solver certain requirements are necessary. First of all the method needs to have a kind of rationale, a justification of what kind of problems can be solved using this method. Next to that it must be clear what kind of products come out of an intermediate step in the process. Every step in the problem solving process, every action, has to have an outcome. These outcomes must be specified in order to provide coaching. Furthermore there has to be a relation between the strategy and the knowledge needed to perform the problem solving task Law students learn about legal concepts in statutes and doctrine, but they do not know how to apply this knowledge to proceed from a specific legal case to an analysis of the problem and an assessment of the situation according to a specific body of norms. Students do not know how to apply their knowledge to a specific case, i.e. how to make the step from support knowledge to operational knowledge. They have to learn to select and qualify facts from a specific situation in legal terms/concepts and link those terms/concepts to rules. There is however a lack of strategy which can give support in the problem solving process. There is no method which tells the students how to proceed, even worse, it is not clear to the student what is expected of her. Students need to gain insight in the task involved, in the knowledge needed to perform the task and in the method or course of actions to take to accomplish the task. Empirical studies with law students solving an administrative law problem show that students do not have a task structure other than a kind of trial and error approach (Muntjewerff, 1993). Research on artificial legal problem solving as performed at my department is the starting point of my research. This research provides a new way to approach legal problem solving and to think about the tasks involved and the knowledge and methods needed for teaching legal problem solving.

Artificial legal problem solving

The research carried out by the department of Computer Science and Law focuses on representation of legal knowledge and on modelling legal problem solving tasks. The research aims at constructing a theory on legal reasoning. The theory is not a psychological theory. The process of legal reasoning of the legal practitioner is not the focus of the research. The focus is on constructing a theory of legal problem solving/legal reasoning, like the approach in (modern) legal theory. The theory under construction is about *artificial* legal reasoning. The research is on identifying the different (abstract) functions in legal reasoning tasks and to develop computational models which realise these functions In research on artificial legal problem solving the models of knowledge representation can be tested by implementing these in a system Research focuses on the abstract functions involved in many legal reasoning tasks and on architectures which support these functions The major assumptions of the theory under construction are :

- In artificial legal reasoning a separation should be made between reasoning about events in the world and reasoning about legal consequences of states in the world.
- The "real" legal reasoning (when no reasoning about the world is concerned) should be viewed as a process of rule application and conflict resolution, rather than drawing logical inferences.

The consequences of these assumptions for knowledge representation are, that in representing regulation knowledge, knowledge about the actions, agents and objects (in the world ruled by the regulation) should be separated from the representation of the regulation. The architecture derived from this view gives the specifications of functions and structure of a legal practitioners workbench to be. The architecture can be enhanced to support various types of tasks of legal practitioners. For example giving advice in cases requires an extension in communication functions which can transform a story into a legally relevant situation description and in planning facilities to optimise legal courses of action (Breuker, 1990, Breuker & den Haan, 1991). The proposed architecture can be enhanced to support the task of training students in a variety of legal tasks (drafting legislation, planning courses of action, argumentation, advice).

Training by means of an Intelligent Tutoring System (ITS)

An ITS is a knowledge based system in which the major function is to coach novices in acquiring expertise in some domain of problem solving. An ITS accomplishes this function by 'looking over the shoulder' of the student who solves a problem presented by the system. The ITS monitors the students problem solving behaviour and compares this with the systems own knowledge about the domain. This means that both the student and the ITS perform the same (sub)task at the same time. When there is a discrepancy between the performance of the system and the performance of the student this is diagnosed in terms of misconceptions or a lack of knowledge at the side of the student. The ITS expertise is the standard. The outcome of the diagnoses triggers a related remedy to correct the misconception or lack of knowledge. To be able to perform the problem solving task an ITS has to embody an expertise component just like a knowledge based system (KBS). However the requirements for expertise to be taught demand more from the model. The way the knowledge is structured should be different and the reasoning process has to be more transparent.

Given the proposed theory on artificial legal reasoning and the consequences the assumptions have on knowledge representation and computational logic, and given the proposed architecture what extensions, refinements or adaptations need to be made when the function of a system is training law students legal problem solving? When the system performs legal reasoning on a task the way proposed in Breuker (1990) and Breuker & den Haan (1991) what kind of adjustments have to be made in order to be able to train legal problem solving to students? In an ITS the system and the student perform on the task at hand at the same time. The way the system reasons, goes about the task and solves the problem, has to be functional for educational purposes. Research on developing a training system for diagnosis in physiotherapy shows that the domain knowledge of an ITS needs to be more explicit and deep (Winkels, 1992). Next to that a model of expertise, constructed in the knowledge acquisition phase of a project, is a descriptive model. The teachability of the strategy however implies a prescriptive strategy, an explicit strategy for solving legal cases that can be learned and executed by students.

The domain of administrative procedural law

The domain under study is the domain of administrative procedural law. The legal sources involved are the General Administrative Law (in dutch de Algemene Wet Bestuursrecht, AWB), case law and other legal texts. In administrative law a set of instruments is provided for local and governmental authorities to accomplish certain objectives. Next to that administrative law provides guarantees for citizens. Public nature and participation are guarantees proceeding a decision of an authority, legal protection is a guarantee afterwards. At the request of one or more citizens, a more or less independent authority checks if the administration follows or followed the rules. In a conflict between a local or governmental authority and a citizen there has to be a judgement by an independent court. However before a conflict reaches the court the administration itself may be asked to give an opinion on the attacked decision of the administration. Generally speaking there are two courses of action for a party concerned :

- to start administrative appeal. This means that a decision taken by an administration will be tested by an other administration (most of the time higher in the hierarchy) on request of a party concerned.
- to make objections. The administrational body which has taken the decision will reconsider the case on request of the party concerned.

In both cases there is no independent court involved, but even so this is also called legal protection. The body or bodies of authority which have to test or reconsider the decisions taken are obligatorily stated in the law. The system is quite complicated. The legislator has stated different factors which determine which procedure has to be followed.

These factors concern the nature of the disputed actions of the authority, the field in which the disputed action was taken, the quality of the party concerned, the type of administration that has taken the decision In the new General Administrative Law (AWB) the procedure of making objections (and ask the administration to reconsider the decision) is a pre-procedure for admission to the court. The recent reorganisation of the courts in the Netherlands has provided new administrative colleges in the law courts. Solving problems in the domain of administrative procedural law involves planning and assessment. Because the exchange of documents between the parties involved plays an important role in the administrative procedure, students need to learn to assess the form and content of the documents involved, need to learn to act and react in the procedure, need to learn to construct the necessary documents and need to learn to plan the course of action. In doing so they need to learn to take the standpoint of the official body as well as the standpoint of the citizen(s)

A case description

A legal case, as used in Dutch legal education, is a written description (a kind of restricted story) of a (potential) problem situation in which some facts and events are mentioned. A link with a certain field of law is already made. A case description is about a half page to a page long and ends with a question which restricts the problem situation and the possible solutions to the problem (the question sets the direction and puts the student on a certain track). The main educational goal of solving a case is (for a student) to learn to handle the theoretical concepts in a specific field of law, to learn to find and apply the specific body of norms, to learn to plan the courses of action and to learn to construct a solution which is legally correct. Administrative law experts and students from the faculty of Law at the University of Amsterdam were engaged in solving the problem below. The thinking aloud protocols gathered in this experiment will be used for (re)constructing the model for teaching legal problem solving. The student protocols will be used for detecting misconceptions and lacks of knowledge.

> Mayor and aldermen of Maastricht decided by degree of 30 July 1981, to grant permission to the Bowl corporation in Maastricht, under clause 56 of the Housing Act, to use the house at Looiersgracht number 12 as an office This is under the condition that the corporation makes suitable for residence (for one and two person households) the office at Looiersgracht number 8 which is in ownership of the Bowl corporation. The Provincial Corporation for Mental Health in Limburg which has an office at Looiersgracht number 8 is very much opposed to the decision of the Mayor and aldermen.
>
> What actions can they take and at which moment?

Domain representation and task model

A teachable model for legal problem solving has to be constructed and decisions on how to teach this model to law students have to be made. A rational reconstruction of the domain knowledge and problem solving strategies are necessary because there is no suitable approach neither in instructional material or in practice. Experts solving problems have an approach which is not suitable for teaching (according to a first analysis made of thinking aloud protocols of experts in the field of administrative law, solving an administrative law problem). An analysis of the types and structures of the domain knowledge is needed. Besides that, an interpretation model from the KADS library of interpretation models is a starting point for modelling the task and the inferences. Regulations consist of abstract norms and definitions. There are also regulations which define authority. Procedural law contains definitions, procedures and courses of action and divides tasks and plans over time. To construct a model of the world actions, objects and agents are distinguished and hierarchies of types are constructed. These typologies are used in reasoning (denHaan & Breuker, 1991). To model the world of administrative procedural law type hierarchies and consist-of relations are constructed.

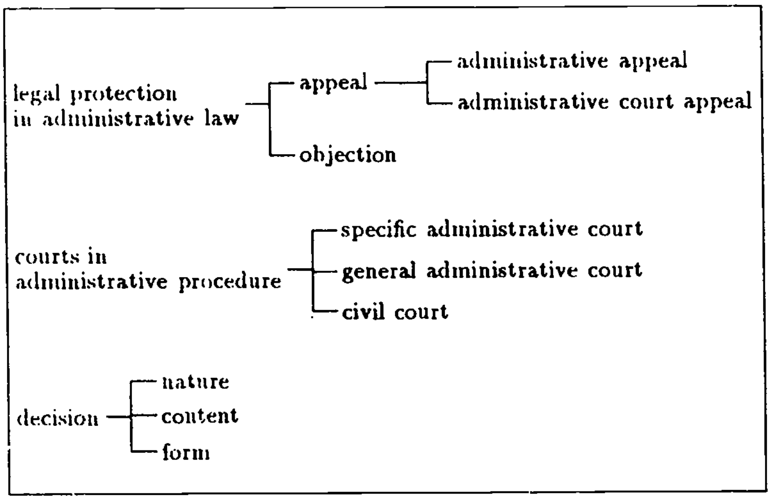


Figure 1: Examples of tree structures in the domain of Administrative procedural law

A most typical task in the domain of law is assessment. A case (situation, events) has to be assessed according to a body of norms. Less frequently occurring tasks are drafting of regulations (a design task) or finding courses of action (a planning task). But even then assessment is implied. For example in administrative law different kinds of documents play a role. First of all a request is made to an authority for a permit. Then a decision is made by the authority. There is a possibility of objecting against the decision. For the party directly involved as well as for a third party concerned. They have to write a complaint and send it to the official body. There is a lot of interaction going on on paper, documents are exchanged, a file grows. This exchange of documents plays an important role in the administrative process. You have to be in time for the right document and write the right kind of objections to be able to proceed. In the process of action and reaction a case is build so to speak. Depending on the state of the documents and the position in the process a certain kind of problem may arise (so a specific case is build) which requires a solution in the form of a decision. In this procedure planning as well as assessment is involved. A conceptual model of assessment is under development in the CommonKADS project as part of the library of interpretation models (Aamodt et al, 1992). An interpretation model is an abstract conceptual model of a set of problem solving methods in terms of inference steps. This model can be used in the process of acquiring knowledge for building artificial legal problem solvers. Assessment tasks are characterized by the fact that a description of a case (situation, events) has to be compared (has to be applied) to norms in order to reach a decision. The task of assessment consists of two subtasks. Case abstraction and norm application. In case abstraction a case description is translated into abstract terms, the relevant items are selected and abstracted. A case description is a structure of events or situations. The abstraction is in most cases identification or classification of instances as concepts. The inference that has to take place is abstraction/transformation. An event or situation in the world has to be abstracted (to be able to compare it to a norm, norms are stated in abstract terms) and/or transformed (to a legally relevant concept, event or situation).

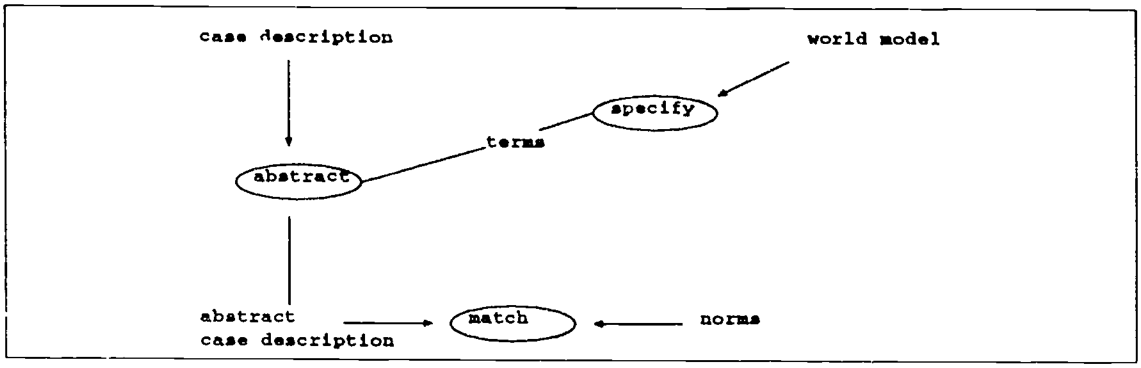


Figure 2: *Legal assessment*

Summary

A representation of the domain knowledge, especially representing the world in which the regulations play a role, has to be performed. The function of the prospective system is education, this means that the knowledge should be structured in a way suitable for teaching and learning. The tasks involved should be modelled. The assessment model could be a starting point to model the task of solving administrative law problems. Thinking aloud protocols of experts in the field of administrative law and of students of the law faculty will be used in this modelling process.

References

Aamodt, A., Bredeweg, B., Breuker, J. Duursma, C., Lockenhoff, Ch., Orsvan, K., Top, J., Valente, A. & van de Velde, W. (1992). *The CommonKADS library*. ECN, SISC, SIEMENS, UvA, VUB.

Abas, P. & Broekers-Knol, A. (1985). *Een methode voor het oplossen van casusposities: een juridisch practicum*. Arnhem: Gouda Quint. (A method for solving legal cases: a legal practical manual).

Breuker, J. & den Haan, N. (1991). Separating world and regulation knowledge: where is the logic? In: *Proceedings of the third international conference on AI and Law. Oxford.*

Breuker, J. (1990). Towards a Workbench for the Legal Practitioner. In: Legal Knowledge Based Systems: Aims for Research and Development. Proceedings of JURIX Conference 1990. Lelystad: Vermande

Crombag, H.F.M., de Wijkerslooth, J.L. & van Tuyll van Serooskerken, E.H. (1972). *Over het oplossen van casusposities*. Groningen: Tjeenk Willink. (On solving legal cases).

Crombag, H.F.M., de Wijkerslooth, J.L. & Cohen, M.J. (1977). *Een theorie over rechterlijke beslissingen*. Groningen: Tjeenk Willink. (A theory on decisions by judges).

den Haan, N. & Breuker, J. (1991). A tractable juridical kbs for teaching and applying traffic rules. In: *Legal Knowledge Based Systems: Model-based legal reasoning. Proceedings of JURIX Conference 1991*. Lelystad: Vermande.

Muntjewerff, A.J. (1993) *Thinking aloud protocols of legal experts and law students solving an administrative procedural law problem*. Amsterdam: UvA.

Winkels, R. (1992) *Explorations in Intelligent Tutoring and Help*. Amsterdam: IOS Press.

The Collaboratory Notebook: a Networked Knowledge-Building Environment for Project Learning

D. KEVIN O'NEILL, LOUIS M. GOMEZ
School of Education and Social Policy and Institute for the Learning Sciences, Northwestern University
2115 North Campus Drive, Evanston, IL 60208
E-Mail: oneill@covis.nwu.edu, gomez@ils.nwu.edu

Abstract: The Collaboratory Notebook is a networked, multimedia knowledge-building environment which has been designed to help students, teachers and scientists share inquiry over the boundaries of time and space. Developed as part of the Learning Through Collaborative Visualization Project (CoVis), the Collaboratory Notebook extends the metaphor of a scientists' laboratory notebook with facilities for sharing inquiry among multiple project partners who may be distributed across institutions, or across the country. Within these facilities, the software provides a supporting structure for scientific dialogue which is tailored to students' attempts to learn about science through project-enhanced science learning.

1.0 The CoVis Project

Science, as it is frequently taught in high school, can be a pretty dull affair. After studying several years of it in the traditional mode, a student may come safely away with barely any impression of what a multifaceted and socially dynamic enterprise it can be.

The Learning Through Collaborative Visualization Project (CoVis) is an attempt to change the way that science is taught and learned in high schools through the use of high-performance computing and communications technologies (Pea and Gomez, 1992; Pea, 1993). In this NSF-sponsored testbed project, our team has adapted technologies used by scientists in the research world for use by high school students in their own classroom research projects. Project-enhanced science learning (Ruopp, Gal et al. 1993) represents a transition from traditional textbook- and lecture-oriented classrooms to ones in which learning occurs in the course of more authentic scientific inquiry. This new mode of learning may help science education to overcome the criticism that it helps develop only inert knowledge that students are unable to apply in appropriate contexts (Brown, Collins, & Duguid, 1989; Hawkins and Pea, 1987; Lave, 1990; Lave & Wenger, 1991).

To support the transition to more authentic project-enhanced learning, the CoVis software suite provides "front ends" to visualization tools used by researchers in the atmospheric sciences which make them usable by 9th to 12th grade students of earth and environmental science (Fishman and D'Amico, 1994; Gordin, Polman and Pea, in press). Also, in order to be faithful to the collaborative fashion in which science is often practiced (Lederberg & Uncapher, 1989), CoVis has provided earth and environmental science students at our two participating high school sites with a variety of means to discuss the research they perform with one another, and with distant researchers and science educators.[1]

[1] CoVis is the first educational application of primary rate ISDN service. Using this new service, local ethernet networks at our two participating high schools are linked to Northwestern University's campus backbone and the Internet. Students and teachers participating in CoVis classes each have personal e-mail accounts which they can use to contact scientists worldwide, and full use of Internet "surfing" tools such as TurboGopher and Mosaic to perform research. Also, to support collaboration between the distant sites, students and teachers make use of the Cruiser™ video and audio teleconferencing system developed at Bellcore, and Timbuktu Pro™ screen-sharing software from Farallon Computing. These two technologies, in concert, allow students to share their project work synchronously across classrooms, and with CoVis graduate students at Northwestern.

One unique tool that we have provided to CoVis students and teachers in this regard is the Collaboratory Notebook. This custom-crafted groupware application is designed to support collaboration on projects across a widely-distributed community of students, teachers and scientists. Sites encompassed by the CoVis community currently include two suburban Chicago high schools, an atmospheric science research lab at the University of Illinois at Urbana-Champaign, and the Exploratorium science museum in San Francisco, California. However, the design of the Collaboratory Notebook permits us to easily extend our community to any locations that the Internet can reach.[2]

Using advanced network technology, we aim to fashion a knowledge-building community that brings isolated classrooms and university labs together in research. In this way, we are attempting to foster a culture in which learning goals and learning resources may be imported and exported freely through classroom walls, creating a "contagion of mind with mind" (Dewey, 1899). We see this as the ultimate extension of the ideal represented by what Perkins (1993) calls a "culture of thinking". By participating in such a culture, students may be involved in ways of working and learning that are currently restricted to university research labs, and may consider scientific careers more seriously than they otherwise would. In any event, their impressions of what "doing science" is like may be more substantial than the rough caricatures that many students depart high school with.

This paper describes the place that the Collaboratory Notebook has within this vision. We begin by outlining some of the design rationale that have driven the initial development of the software, including assumptions made by the researchers about the classrooms in which it would be used and the way that project-enhanced science learning and teaching would be practiced there. We then attempt to lay out the lessons that we have learned during the early days of the software's use by students, teachers and scientists, and discuss more broadly the implications of these lessons for the development and use of groupware for school settings.

2.0 A Brief Introduction to the Collaboratory Notebook

Most simply, the Collaboratory Notebook is a structured hypermedia database in which students and other CoVis community members can record and elaborate upon project ideas and efforts. These recordings take the form of text and annotated images such as charts, graphs, satellite photos and other scientific visualizations that are generated in the course of a project.

The organizing structure of the Collaboratory Notebook's database is built upon a library metaphor; with a bookshelf, notebooks, and pages being the primary interface elements. When a student, teacher or scientist logs on, his bookshelf displays all of the notebooks which he is permitted to read and write in.

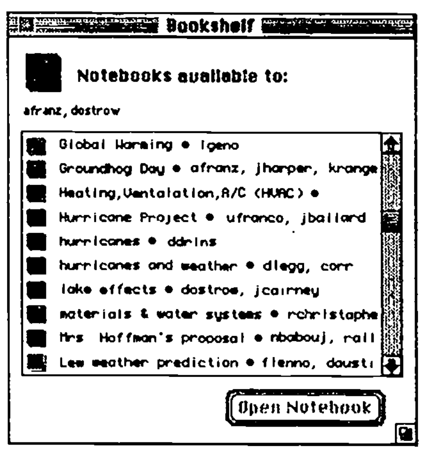


Figure 1. A Bookshelf. The color and letter on the icons at left indicate whether the notebook is a private journal (J), group project notebook (P) or public discussion (D).

[2] The Collaboratory Notebook is a Macintosh application which runs on 7 Quadra 700s at each of our participating high schools. This application functions as a client to an Oracle database maintained at Northwestern University on a Sun Sparcstation, which stores all notebook data. Communication between each Macintosh and the database server takes place via TCP/IP using Oracle's SQL*Net driver for the Macintosh.

Any user may create three different kinds of notebooks. These are:

Private Journals. Only the author (creator) of a private journal can read or write in it.
Group Project notebooks. A project notebook can have one or more authors. Only its authors are permitted to read or contribute to it.
Public Discussions. A discussion is a public notebook open for any member of the CoVis community to read and contribute to.

Within each notebook are an arbitrary number of pages, written by its authors. These can be organized in a variety of ways, from a simple sequential ordering (like that of Usenet newsgroups) to rather elaborate interconnected webs.

2.1 Writing in a Notebook

In order to help students organize and make sense of their efforts while working on science projects, the Collaboratory Notebook encourages them to write in small chunks, referred to as "pages". These pages are organized using hypermedia links that describe the relationships they bear to one another in a vocabulary of scientific dialogue.

When a notebook author or authors wish to write something new, they open a notebook, and create a new page within it. A "page writing window" then pops up on the screen, with blank areas for a page title and body text. There is also a button which provides a means of attaching pictures to the page.

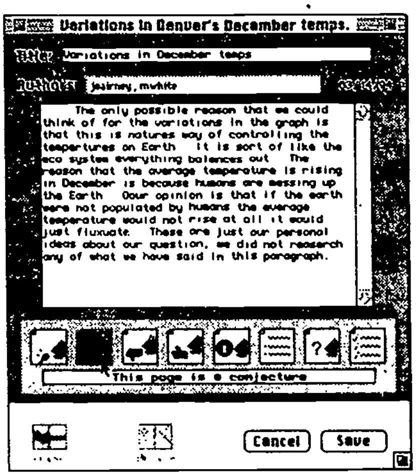


Figure 2. A new page being written by two authors.

The band of icons near the bottom of the page writing window provides authors with an array of possible labels to describe what they write. At any time in writing a page, the authors choose one of these labels. The software allows them to change their minds repeatedly in the course of their work, if their sense of what they have written changes; however, the computer does not save the page until some label has been chosen.

The set of page labels that we are presently experimenting with consists of:

- *information*
- a *commentary* on someone else's writing
- a *question*
- a *conjecture* (possible answer) for some question
- *evidence for* or *evidence against* a conjecture
- a *plan* for action
- a *step in a plan*

If a new page has been written in response to another page in the notebook, the software creates hypermedia links between the new page and the previous one, using the labels that their respective authors have given them. All of the choices of page labels are available to an author creating an entirely new page,

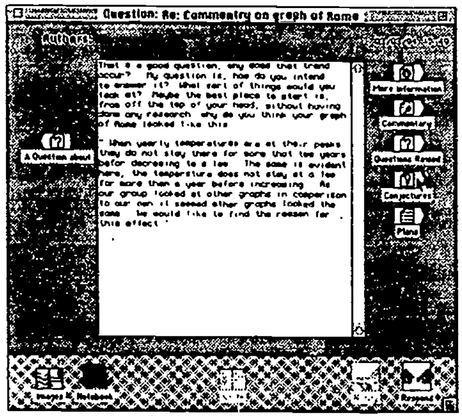


Figure 4. A notebook page being read

however an author writing a response to an older page is limited to a subset of these labels, according to the to the semantics of the label chosen for the page he is responding to. These limitations are put in place as a framework that encourages articulation of crucial steps in an inquiry.

For example, an author may not link a new page as evidence for or against a question, since a question does not put forward an explicit claim. However, an author may to respond to a question with a conjecture about a possible answer, and conjectures may in turn be responded to with evidence, since they imply predictions. In a similar fashion, a plan of action is not allowed as a response to the noncommittal "information", but is permitted as a response to a conjecture or a question, which more explicitly call out for investigation.

Page Type	Response Palette
Question	Information, Question, Commentary, Conjecture, Plan
Conjecture	Information, Question, Commentary, Plans, Evidence For, Evidence Against
Plan	Information, Question, Commentary, Step in a Plan

Figure 3. Response palettes available for question, conjecture and plan pages.

These varying sets of response palettes for pages of different types represent a kind of scaffolding for project-related dialogues. This dialogue scaffolding has embedded within it a simple research strategy: "begin with a question, project possible answers, develop a research plan, and follow each step through". In its gross structure and purpose, it bears some affinity to ideas such as Collins and Ferguson's (1993) "epistemic forms and epistemic games", and Perkins' (1986) "thinking frames"; though our emphasis here is more directly on facilitating a learning dialogue between collaborators than on an individuals' thinking.

2.2 Following Discussions in a Notebook

Notebook readers can follow threads of dialogue from page to page using link buttons that appear on the right and left of each page window (see Figure 4). The right-hand arrow buttons allow browsing forward in the web of dialogue, while the left-hand button allows browsing backward. The color of these buttons, which bear the names of the response categories for a page of the given type, change to indicate how many responses of each kind have been made to that page so far.

These colored link buttons serve two purposes. First, they show the "beaten paths" that others have taken in responding to each page, reflecting a kind of social consensus. Second, they serve as prompts for further writing. For instance, when a student reads a conjecture page, its display reminds him that in addition to raising questions about it, commenting, or adding more information, he might respond by providing evidence for or against it. Similarly, when he is reading a question, he is reminded that he could propose a conjecture about it or a plan to research an answer.

A broader overview of the dialogue taking place in a notebook is provided by its table of contents (see Figure 5). This window shows the title and authorship of the notebook, and provides a scrolling list of its pages which can be sorted in several ways.

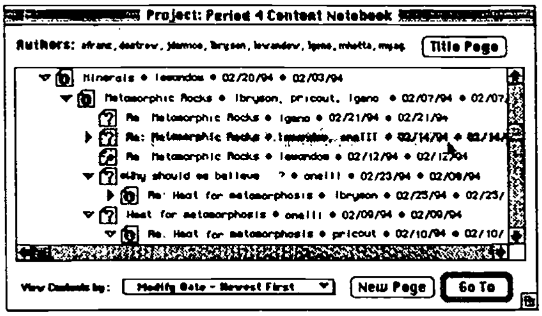


Figure 5. A notebook table of contents

Each line in the list of pages begins with an icon which indicates what type of page it represents. To the right of these are its title, a list of the authors who wrote it, and the date that they wrote it on. If a page has been responded to, it appears with a twist-down triangle to the left of it. Clicking on this triangle displays the immediate responses to that page, indented beneath it. More deeply indented pages are ones written further along in the chain of dialogue; thus, the table of contents for a notebook allows a reader to view, in diagrammatic form, any strand of conversation that has taken place within it. By clicking on any one of the pages, he can also jump directly to it without navigating from page to page.

3.0 Designers' Assumptions and Expectations of Use

Now that we have said a little about how the Collaboratory Notebook works, we would like to say something more about why we designed it as we did, and how we expected it to fit into its intended settings. At this point, we must outline a few of our theoretical perspectives on science learning. The most important of these are:

- The practice of science has important social dimensions which deserve a place in high school science learning. Learning about a science ideally involves a process of "enculturation" into a community of its practice, in the sense used by Perkins (1993).
- The process of enculturation into a community of practice includes the appropriation of some of the language which customarily accompanies and facilitates its tasks. Since language is acquired only through use (Bruner, 1990), students' articulation of their ideas and the process of their learning while they work is crucial to enculturation. (See Collins, in press, for further benefits of articulation.)
- The effort to reason purposefully in collaboration with others, especially in writing, provides the best possible motivation and occasion for articulation.
- Open and purposeful dialogue with respect to a shared goal is not, in itself, scientific. Rather, science has a unique speech genre of its own (Wertch, 1991), and science learning should provide students with experiences that will enable them to "find a voice" within it (Belenky et. al, 1986).
- Some elements of the scientific speech genre can be embodied within electronic media in such a way that they facilitate both the acts of investigation and the process of finding a voice within this speech genre.

In light of these theoretical perspectives, particular attention has been paid in the design of the Collaboratory Notebook to facilitating project-centered dialogue. This attention is reflected especially in two facets of the software.

Page Labels. We expected that our small, fixed set of page labels would have three significant advantages for the CoVis community. First, we wanted to draw students' attention to the small intermediate products that are required in the course of performing investigations. We believed this would be helpful to teachers and scientist mentors by providing significant milestones for them to use in tracking students' progress within a project.

Second, we expected that the conversations that students and teachers would have both within, and in reference to the Collaboratory Notebook would make explicit the privileged status given to questions, conjectures, research plans and evidence in scientific dialogue, and provide students with an opportunity to find a voice within this speech genre.

Last, we felt that a limited range of page types and possibilities for linking them would encourage the development of consistent conventions of use across the large, distributed community that CoVis supports. This, in turn, would facilitate remote collaboration and the fullest use of the intellectual resources of the community.

Tables of Contents. We anticipated that one of the teachers' and scientist mentors' greatest needs within the Collaboratory Notebook environment would be a means to rapidly determine the state of a project effort. In order to be effective coaches of project science, we felt that teachers and mentors would need to quickly determine where a project effort was headed, what had been accomplished, how labour was being distributed by the participants, and so on. By logging on and viewing the tables of contents for students' project notebooks, we expected that teachers and scientist mentors would be able to get a quick overview of what had been accomplished by a particular project team, without asking them to submit a report or make a formal presentation.

Besides the presence or absence of particular types of pages, a notebook's table of contents can indicate other things about a project effort that a coach might find useful to know. For example, if some group action, like a novel weather visualization, graphic, or illustration has been recorded, one may ask, was it followed up with research questions? Were these the sort of "wonderment" questions that seem likely to lead to new learning (Scardamalia and Bereiter, 1991), or merely the sort that one might answer by leafing through an encyclopedia? If wonderment questions were posed, were they followed up with any conjectures that could be proven or disproven with available or collectable data? And were plans made for collecting such data?

4.0 Lessons from Early Use of the Collaboratory Notebook

Though the software has only been available for use in our classrooms for five months at the time of this writing, some of our expectations about the purposes that it would serve have already been given significant testing. Four CoVis teachers have had their students use the software in a variety of project contexts, spanning roughly 45 hours of class time in total over the first three quarters of the 1993/1994 school year. In this time, 118 students in 6 classes have created 59 notebooks containing a total of 352 pages.

Our current design of page labels and links has met with mixed success. In at least two projects, this design seems to have helped teachers to make work assignments involving collaboration clear to their students. By sketching or modeling the hierarchies of notebook pages that they would like their students to create, teachers can describe in rough outline the kinds of dialogues that they see as suitable to a particular research purpose. This allows them to provide some high-level guidance to their students with respect to dialogues between student collaborators, and among student collaborators and remote mentors.

Our expectation that a small set of page types would help in establishing conventions for the use of Collaboratory Notebook seems to have been partially validated as well. In classroom uses so far, we have seen customs of work arise that involve specific forms of page hierarchies to support dialogues for a variety of ends. For example, teachers have invented forms to support the revision of successive drafts of students' project proposals, the organization of knowledge gleaned by students from interviews with experts, the pooling of knowledge about weather prediction, and the sharing of responsibility between a teacher and a graduate student for mentoring groups of students working on projects.

To this point, however, the range of page labels in our experimental set has not been used to its fullest. Although students and teachers took very easily to using the information, question, commentary and plan labels to compose simple webs of project-centered dialogue, it is taking some time for them to develop more elaborate customs that involve conjecture and evidence pages in a way that is central to these conversations. This, we believe, is a reflection of many things: of (1) the fact that both our teachers and students are new to the project-enhanced method of science learning, (2) the need for new assessment strategies to complement the new modes of learning, (3) the practical limitations of our software and the technical setting in which it is used, and (4) our own difficulty in communicating the purposes and potential uses of the software clearly to teachers at an early stage. We will attempt to explain each of these now in turn.

Though the ultimate goal of building CoVis' distributed learning environment is to make the teaching and learning practices of high school more similar to those of the research lab, classroom practices in a public school science classroom arise from a marriage between this tradition and its own, very different one. Part of the difficulty that our teachers and students are facing in their adoption of the Collaboratory Notebook seems to stem from the sharp disjunction between the inquiry-centered dialogue that is privileged by our software's page labels, and the assessment-driven dialogues that traditionally take place in classrooms. On the whole, our teachers have been loath to assign any work to which they cannot confidently attach a grade value, and their students have been quite opposed to doing such work. Since our teachers are only just beginning to adopt the kinds of metrics that will help them to assess the work done by their students' using the Collaboratory Notebook, their efforts to make better use of the tool are likely to move in step with this initiative.

In light of the strong bond between assessment and adoption, it may be useful for us to alter our design to incorporate special forms of pages for project proposals, delivery contracts between teachers and students, and even interim status reports from teachers. In appropriating the software to their current uses, teachers and students have frequently used our "information" pages for purposes like these.

Our tables of contents proved to be a good start for a project monitoring and assessment aid, but will clearly not be adequate to this purpose in the long run. It has proven more time- and labor-intensive than we had anticipated for teachers to track student progress on projects, primarily because a notebook's table of contents does not provide them with a simple and efficient way to track change from day to day or week to week. Before the start of the next school year, we plan to implement a set of tools designed specifically to aid in project monitoring. These will need to be designed in close consultation with teachers.

In the future, we will also need to be more concerned to provide our teachers with ways to review their students' work at home, as they are accustomed to doing with paper documents. Since our software currently makes large demands on both system RAM and the speed of its connection to our database server, our teachers are not all currently equipped to use the software from their homes using the university SLIP facilities. As a partial solution to this problem, we recently implemented a new software feature that allows teachers to download a notebook's entire contents to a floppy disk, and read it at home with a standalone version of the application. However, this does not provide help to teachers who do not have computers capable of running our software.

5.0 Challenges for the Design and use of Groupware for School Settings

In addition to those stated above, three more general lessons that the CoVis team has learned this year deserve special attention by developers and users of future systems to support collaborative learning over distance.

First, in order to reap the pedagogical benefit of technologies for asynchronous collaboration, one must take care only to apply them in projects for which collaboration is needed. The coordination of multi-party efforts over time must be integral to students' efforts in producing the results desired for a project, or the challenge of the collaboration itself will only serve as an annoyance. In our experience, collaboration is not considered a benefit by high school students unless it is directly useful to completing the work that is assigned them. Thus, if one attempts to add an element of collaboration to students' traditional work in an effort to make it more motivating, the results are very likely to disappoint.

Second, the network connections and servers that support .. distributed, multimedia learning environment like that of CoVis must be very well maintained and *very* reliable. Because class periods are usually quite short, even a five minute interruption in service at an inopportune moment can throw teachers' and students' plans into an unrecoverable tailspin; and a loose ethernet connector in the wrong place can degrade the performance of an application like the Collaboratory Notebook significantly. If network resources are relied upon to a degree that will justify their cost, such failures can generate significant distrust and disuse of the technology—even among the most technically savvy teachers and students.

Finally, since teachers often plan the gross structure of their curricula far in advance, it is extremely important for developers to communicate the purpose and promise of groupware tools as early on as possible, in a way that teachers can clearly relate to their own curricular goals and classroom practices. In an environment like the CoVis testbed, where teachers and students have access to a large variety of electronic media including email and netnews, fitting a tool to a purpose can be difficult. For this reason, the special uses of the different tools need to be clearly set out, and consultants must be available with whom teachers can discuss and develop their plans to use them.

References

Belenky, M.F., et al. (1986). Women's ways of knowing: The development of self, voice and mind. New York: Basic Books.

Brown, J.S., Collins, A. and Duguid p. (1989). Situated cognition and the culture of learning. Educational Researcher 18(1), 32-42.
Bruner, J. (1990). Acts of Meaning. Cambridge, Mass: Harvard University Press.
Collins, A. (in press). Design issues for learning environments. in S. Vosniadou et. al (eds.) International perspectives on the psychological foundations of technology-based learning environments. New York. Springer-Verlag.
Collins, A. and Furguson, W. (1993). Epistemic forms and epistemic games: structures and strategies to guide inquiry. Educational Psychologist, 28(1), 25-42.
Dewey, J. (1899). Lectures in the philosophy of education. New York: Random House.
Gordin, D.N., Polman, J. and Pea, R.D. (in press). The climate visualizer: Sense-making through scientific visualization. To appear in the Journal of Science Education and Technology.
Hawkins, J., & Pea, R.D. (1987). Tools for bridging everyday and scientific thinking. Journal for Research in Science Teaching, 24(4), 291-307.
Brown, A.L., Ash, D., Rutherford, M., Nakagawa, K., Gordon, A., and Campione, J.C. (1993). Distributed expertise in the classroom. in Salomon, G. (ed.) Distributed Cognitions. New York: Cambridge University Press.
Lave, J. and Wenger, E. (1991). Situated learning: legitimate peripheral participation. Cambridge University Press.
Lave, Jean (1993). The culture of acquisition and the practice of understanding. In J.W.Stegler, R.A. Sweder and J.H. Herdt (eds.). Cultural psychology: Essays on comparative human development. Cambridge University Press.
Lederberg, J., & Uncapher, K. (1989). (Co-Chairs), Towards a National Collaboratory: Report of an Invitational Workshop at the Rockefeller University, March 17-18, 1989. Washington DC: National Science Foundation Directorate for Computer and Information Science.
Pea, R.D., & Gomez, L.M. (1992). Distributed multimedia learning environments: Why and how? Interactive Learning Environments, 2(2), 73-109.
Pea, R.D. (1993, May). Distributed multimedia learning environments: The collaborative visualization Project. Communications of the ACM, p. 60-63.
Perkins, D.N. (1986). Thinking Frames. Educational Leadership 43, 4-10.
Perkins, D.N. (1993). Creating a culture of thinking. Educational Leadership 51(3), 98-99.
Ruopp, Gal, Drayton and Pfister (eds.) (1993), LabNet: Toward a community of practice. Hillsdale, NJ: Lawrence Erlbaum Associates.
Scardamalia, M. and Bereiter, C. (1991). Higher levels of agency for children in knowledge building: A challenge for the design of new knowledge media. Journal of the Learning Sciences 1(1), 7-68.
Wertch, J.V. (1991). Voices of the mind. Cambridge, Mass: Harvard University Press.

Acknowledgments

The writers would like to thank Joey Gray for his invaluable aid in programming and designing the Collaboratory Notebook. We also gratefully acknowledge Scott Storkel of Oracle for his help in developing much of the code used to access its underlying database. Further thanks are due to Roy Pea and Danny Edelson for their helpful comments on drafts of this paper, and to the National Science Foundation for its support of this work under Grant MDR-9253462.

Transfer of a Natural Language System for Problem-Solving in Physics to Other Domains

GRAHAM E. OBEREM
Department of Physics FM-15, University of Washington, Seattle, WA 98195, U.S.A.
E-Mail: oberem@u.washington.edu

Abstract: A natural language system originally designed to understand kinematics problems has been transferred to different topics in Physics and to subject areas as diverse as Logic and Roman Law. These intelligent tutoring systems are able to understand plain English text and can engage students in a natural language dialogue. Although its capabilities are limited, the portability of this natural language system and the advantages it offers over traditional approaches to answer judging in computer assisted instruction make it an attractive alternative for many CAI applications. This paper describes the language processing system and several programs that use it.

INTRODUCTION

For many students, problem-solving in physics consists of a matching formula with information given in a problem statement (Chi, Feltovich, & Glaser, 1981). To some extent this can be done without a proper grasp of the physical concepts involved, but this can be very frustrating for students. Ideally, problem-solving should reinforce the concepts that have been learned in class. Coaching a student in problem-solving involves dialogue and such dialogues are often Socratic in style (Mestre, Gerace, Hardiman, & Lochhead, 1987).

It has been suggested that the computer could be used to personalize problem-solving instruction, even in large classes, but the limited language capability of CAI systems has made it difficult to achieve this goal (Trowbridge & Chioccariello, 1985). Patrick Suppes (1990) stresses the need for adequate language processing in CAI in general, and some intelligent tutoring systems have attempted to address this issue (Brown, Burton, & de Kleer, 1982). However, the majority of CAI programs, even intelligent tutoring systems, still avoid the problem by constraining the user to menu selections, keyword answers, or multiple-choice questions.

The intelligent tutoring system, ALBERT, is a problem-solving monitor and coach that has been used with high school and college level physics students for several years (Oberem, 1987). ALBERT uses a natural language system to understand kinematics problems stated in plain English. The program not only understands physics problems, but also knows how to solve them and can teach a student how to solve them by engaging the student in a plain English dialogue.

ALBERT was originally implemented on a mainframe computer to obtain the speed required for the language processing and problem-solving. However, advances in microcomputer hardware and software over the last few years have made it possible to move ALBERT to the PC and Macintosh platforms. Due to the modular construction of ALBERT, it has been possible to use the natural language system and the expert problem solver in several new programs for teaching physics and other subjects. In this paper we discuss four programs that incorporate the natural language system developed for ALBERT.

THE NATURAL LANGUAGE SYSTEM

The natural language system developed for ALBERT is intended to augment existing CAI tools. It is not a generic natural language processor. Its robustness in the programs described here arises out of the fact that it is called upon to operate only in a very limited domain of expertise for any given application. In the case of ALBERT, the language system is only used to understand textbook kinematics problems and student-computer dialogues about kinematics. ALBERT's language system uses a syntactic pattern matching technique to parse the input sentences and includes semantic routines to complete the process of understanding. It also has the ability to deal with syntactically incomplete sentence fragments, a feature required for dealing with the vagaries of the more colloquial student-computer dialogues. The departure from the keyword matching approach of traditional CAI enables ALBERT to engage students in a fairly natural problem-solving discussion.

Figure 1 illustrates the major components of ALBERT's language processing system. The vocabulary of one dimensional kinematics is very limited and largely free of ambiguity. Words in the lexicon are therefore grouped into approximately twenty five categories based on their semantic function in this domain of physics. Traditional linguistic groupings such as noun and verb are not used. In the first pass, the words in the input text are parsed into a numerical string where each number in the string corresponds to the lexical category of the associated word. Regular patterns occur in these numerical strings and these patterns correspond to syntactic entities containing important information.

A knowledge-base of approximately one hundred commonly occurring syntactic patterns is maintained in ALBERT. This list of patterns was obtained by analyzing the syntax of textbook physics problems and student-computer dialogues. In the second phase of the parse, the numerical string derived from lexical analysis is searched for these syntactic patterns. When a particular pattern is identified in the input string, a parameter-driven semantic routine is called to extract the information from that part of the sentence. Very few semantic routines are needed, only five in the case of ALBERT. When semantic processing is complete, all the information in the typed input will have been used to instantiate variables in computational models of the kinematics expert , the student, and the tutorial process.

In addition to the natural language processor, ALBERT includes a generic expert problem-solver, a tutorial model that is based on the results of research on the teaching of problem-solving (Reif, 1981), and a tutorial management system that is used to guide ALBERT's interaction with the student. Although all of these components are modular, the language processor has been of the greatest value in the development of new intelligent tutoring systems requiring natural language capability.

MOVING THE LANGUAGE SYSTEM TO A NEW DOMAIN

Moving ALBERT's natural language system to a new subject area is a relatively simple programming task, but it does require a significant amount of effort in analyzing the language of the new domain. Three components of the language system need to be modified for a new application: a) the lexicon, b) the syntactic pattern database, and c) the semantic routines. The first two take the form of alphanumeric data. Modifying them is a straightforward editing task. The third component comprises a small collection of sub-routines, some of which are generic and can be transferred to new programs with only minor modification. For example, the routines that extract and store the numerical values of physical quantities can be used with little or no change. Modifying the semantic routines or producing additional ones is a programming task the complexity of which is determined by the nature of the target domain.

Developing a new lexicon involves analyzing examples of the language that will be typical in the new area of interest. Each word must be classified according to its semantic function and added to the appropriate group in the vocabulary. A given word may be grouped differently for different applications and, in new programs, it has been found that new groups need to be added to the lexicon while some of the old ones can be discarded. The vocabulary can be refined and expanded once the program is in use.

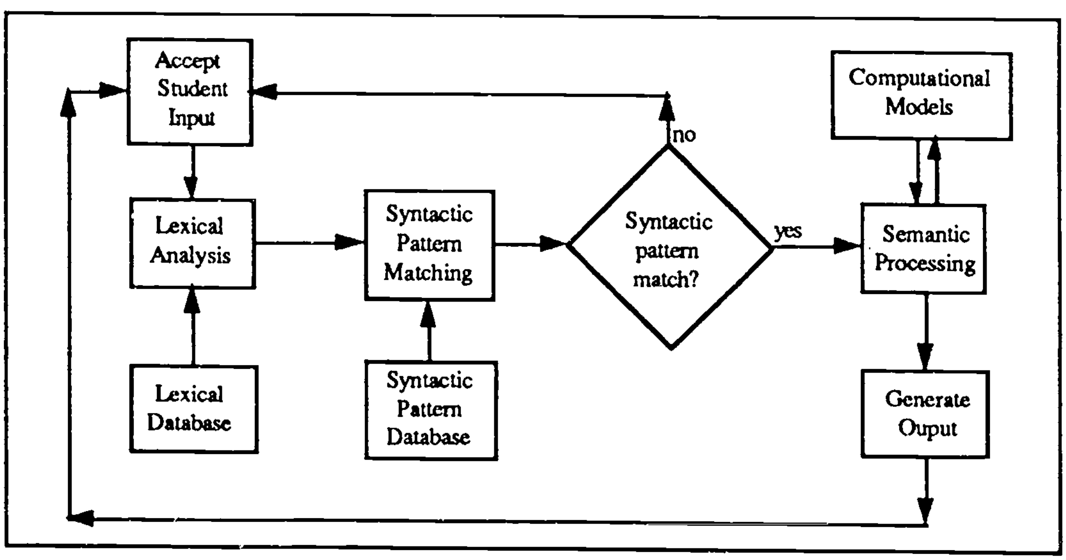


Figure 1. The structure of ALBERT's natural language system

The commonly occurring syntactic structures are also identified by analyzing examples of text from the new domain. These must be encoded as numerical patterns to match the word classifications in the lexicon. A small number of data elements are associated with each syntactic pattern to act as parameters for the appropriate semantic routines. Inevitably, additional syntactic patterns will be identified during the initial testing phase and can be included at that time.

The examples that follow show how the process just described has been used to produce intelligent tutoring systems that incorporate a natural language interface.

FREEBODY

Research has shown that students have a great deal of trouble learning some of the most basic concepts in introductory physics (McDermott, 1991). One way of helping students to understand these concepts is to engage them in a qualitative dialogue about carefully chosen physical situations. This often requires students to draw and discuss diagrams. In addition to the need for language processing, one must address the problem of understanding student diagrams drawn on the computer. Although research is providing the techniques needed for interpreting text and graphics in complex physical situations (Novak & Bulko, 1993), the problem is often simplified by the unambiguous nature of the context, for example, when students are required to draw vectors.

Many students have trouble identifying the forces exerted on a given object (McDermott, 1984). A "free-body diagram" is a formal representation in which the forces on an object of interest are shown as vectors. The student is expected to identify the type of each force and the agent producing the force. Some students find it difficult to identify all of the forces. They propose non-existent forces and often cannot identify the agent producing a given force. FREEBODY allows students to draw free-body diagrams on the computer with a mouse. The program discusses each of the forces that the student draws and helps the student to correct any misconceptions that he or she might have. FREEBODY also assesses whether or not the resulting diagram is reasonable in terms of the situation described in the problem statement. For example, in cases where the net force on an object is zero, the sum of the force vectors drawn by the student should be zero.

Figure 2 shows a typical screen display from FREEBODY. The problem is displayed in the upper left quadrant of the screen. A summary of the forces is maintained in the lower left quadrant. Students can refer to that area of the screen for information about forces already drawn and described. Students draw the free-body diagram in the upper right part of the screen and the dialogue takes place immediately below that. The flexibility of the intelligent tutor permits the student to draw the force vectors in any order and to adjust the length and orientation of a vector during the discussion. The program interprets both plain the English dialogue

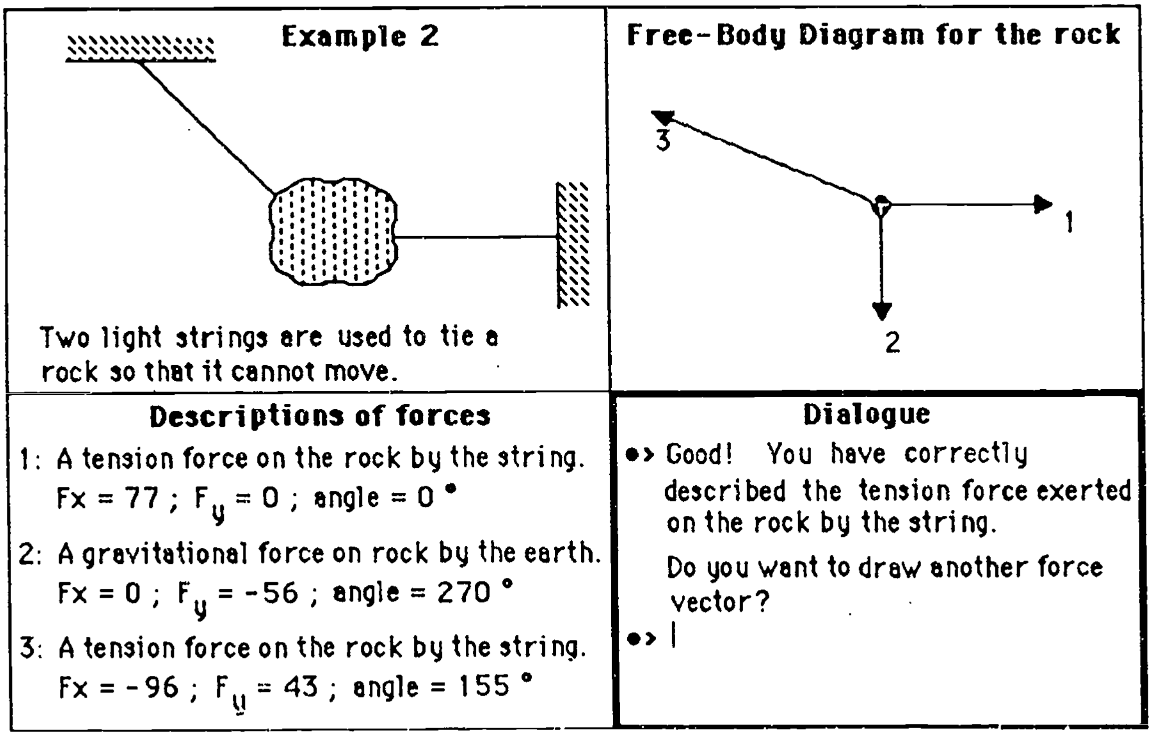


Figure 2. The screen layout for FREEBODY.

Example 1:	A man is pulling on a rope attached to a box. The box does not move. Draw a free-body diagram for the box.	*[Man shown pulling to right on rope.]*
Freebody:	What kind of force is force #2?	*[Student draws vector to the right.]*
Student:	a pulling force by the person on the box	
Freebody:	What kind of force is a 'pulling' force?	
Student:	a force of motion	
Freebody:	What type of force is a 'motion' force?	
Student:	a contact force	*[Incorrect. The man is not in contact with the box.]*
Freebody:	What kind of contact force is it?	
Student:	a tension force by the person on the box	
Freebody:	A tension force is a contact force. The person is not in contact with the box. Can the person exert a tension force on the box?	
Student:	yes	
Freebody:	I don't see how that can be. A contact force implies contact between the objects. So what is exerting this tension force which you drew?	
Student:	the string exerts the force on the box	
Freebody:	Good! You have correctly described the tension force exerted by the string on the box.	

Figure 3. An extract from a student dialogue with FREEBODY.

and the graphics input. FREEBODY has been used extensively in the introductory calculus-based physics courses at the University of Washington for the last two years. All the dialogues are recorded in full and this allows us to use the program as a research tool to probe student understanding of force (McDermott, 1990). Figure 3 is an extract from a discussion between FREEBODY and a physics student that shows this student's difficulty with contact forces. We are currently investigating the effectiveness of this program in dealing with such misconceptions (Oberem, Shaffer, & McDermott, 1993).

In developing FREEBODY it was found that fewer words were required in the lexicon than was the case for ALBERT. We also found that the syntax used to describe the forces could be matched with fewer syntactic patterns than were needed for ALBERT. The semantic routines were restructured to deal with the more qualitative data, since there is no numerical problem-solving in FREEBODY. The ease with which ALBERT's language system could be adapted made the development of this program relatively straightforward.

ILONA

ILONA is a program that helps students translate logical statements into mathematical form. The student types in a logical proposition and then "writes" it in mathematical form by clicking successively on the appropriate symbols that appear on the screen. This program is different from ALBERT and FREEBODY in that there is no dialogue as such. The natural language system is used to understand logical sentences such as: *If all cheese is blue then some grass is green.* ILONA can translate such input sentences into mathematical form and explain its reasoning by stepping through the operations in sequence. Figure 4 shows how ILONA would solve and explain the example above. ILONA is also able to check the student's answer and give specific feedback about errors which the student might have made. However, the degree of flexibility given to the student in the choice of variable names and function names makes a literal comparison of ILONA's answer with that of the student difficult. To overcome this problem, the student's answer is parsed into the canonical internal representation used for ILONA's answer. Answer judging is then relatively straightforward, as is the ability to provide specific feedback on errors. Figure 5 shows how ILONA can assist the student who has made an error.

The natural language system used in ILONA was also derived from ALBERT. In this case, the sentence structures were found to be very limited, thus requiring very few syntactic patterns. However, the vocabulary is extensive due to the unlimited range of topics. In fact it is impossible to include every conceivable noun and adjective in the lexicon of a CAI program like ILONA. Accordingly, ILONA's version of the language system was modified to cope with unrecognized words. Consider the sentence: *If it is the case that all big whatzits are flubsy, then some small thatzits are wubsy.* Clearly words like *whatzits* and *thatzits* will not be found in any dictionary. However, we can infer from the sentence structure that these two unknown words are probably

Type in a logic proposition: If all cheese is blue then some grass is green.

ILONA would have worked it out like this:

Let the variable "x" represent an element of "cheese"
Let the function "f( )" represent the property, ability or quality associated with "blue"

Let the variable "y" represent an element of "grass"
Let the function "g( )" represent the property, ability or quality associated with "green"

Therefore: $\forall x\ f(x) \longrightarrow \exists y\ g(y)$

Figure 4. An example of how ILONA solves a logic problem.

Type in a logic proposition: some animals are brown or white

Student answer: $\exists y\ (h(y) \wedge k(y))$

There is a problem at this point. . . . the "and" should be "or."

Click on ERASE and correct your answer.

Figure 5. An example of the feedback ILONA gives when a student makes an error.

nouns, and that the words *flubsy* and *wubsy* describe some property of these nouns. The natural language system in ILONA deals with unrecognized words by inferring their most probable function in the sentence from the sentence structure. This "fuzzy" parsing appears to work well for ILONA.

ILONA is part of a broader CAI tutorial that teaches the fundamentals of logic. ILONA provides students with a flexible work environment within which to practice what they have learned. The program is discussed more fully in Oberem, Mayer, & Makedon (1992).

CICERO

CICERO is a program that was developed as part of a series of lessons in the field of Roman Law (Lambiris, 1986). It allows students to participate in a simulated lawyer-client interview. The student, acting as a lawyer, asks the client questions to ascertain the facts of the case. The student then applies his or her legal knowledge to give advice to the client on the probable outcome of the case. A natural language processor is required to achieve a realistic and educationally rewarding flow of dialogue between the student-lawyer and the computer-client. It is important to allow a random sequence of questions, composite questions, pronouns and other linguistic devices common in the legal profession. The student must be allowed to ask questions in ordinary language, using alternative sentence structures and terms where appropriate.

The nature of the Roman Law of marriages is such that there is a limited range of relevant facts that are pertinent to the cases posed in the exercises. The relevant facts include such things as the age of the parties, whether they are related by blood and to what degree, their willingness to marry each other, and so on. It would not be relevant to ascertain what either party's tax liability is, or what their preferred sporting activity might be. Another factor which renders language understanding more tractable in this situation is the relative predictability of the form of the expected input. Input will be in the form of questions designed to elicit specific factual responses. The number of possible questions is thus finite as is the scope of the vocabulary that a lawyer would use. ALBERT's language system is well suited to this type of application.

The vocabulary needed for this program had been studied in an earlier attempt to design a lawyer-client simulation using conventional CAI answer-judging techniques. It was thus relatively easy to form the required lexicon by grouping the words according to their semantic function in the lawyers' questions. It was also

Student:	How old is Quintus? Also are you of senatorial rank?
CICERO:	Quintus is 24 years old
	I am not of senatorial rank.
Student:	Why?
CICERO:	I am not a senator and neither is my father.
Student:	Were you ever a slave and is Quintus a senator?
CICERO:	I am freeborn.
	Quintus is of senatorial rank.
Student:	Are you both citizens?
CICERO:	Quintus is a citizen.
	I am a citizen.
Student:	Who is your paterfamilias and does he consent to the marriage?
CICERO:	My paterfamilias is my father.
	My paterfamilias is agreeable to the marriage.
Student:	Is Quintus alieni iuris and if so, does his paterfamilias consent?
CICERO:	Quintus is dependent.
	His paterfamilias is agreeable to the marriage.
Student:	Who is his paterfamilias?
CICERO:	His paterfamilias is his grandfather.
Student:	Are you a widow, is Quintus a castratus and are either of you actors?
CICERO:	I am a widow.
	Quintus is not a castratus.
	Quintus is not an actor.
	I am not an actress.

Figure 6. An extract from a dialogue with CICERO.

recognized that a relatively small set of syntactic structures are common to many of the questions asked. A single semantic sub-routine suffices to transform the input questions into one or more arrays of pointers that index into a knowledge-base of information about the two parties. This approach had been developed previously in a program designed to engage English second language (ESL) students in a computer-based dialogue (Oberem & Mostert, 1990). The answers given by CICERO are constructed using a very simple language generator that forms English sentences by placing pronouns, verbs, articles and other words around information contained in the knowledge-base. Each piece of knowledge in the database can also have an explanation associated with it. This allows students to ask "why" questions as illustrated in the dialogue shown in Figure 6.

CICERO keeps track of a student's progress during the course of the discussion. This information is used to judge whether or not a student has uncovered enough facts to give reliable advice and also enables CICERO to offer assistance to those students who have difficulty asking appropriate questions. Lambiris and Oberem have studied the use of this program and its earlier more traditional counterpart (Lambiris & Oberem, 1993). ALBERT's language system provides a more realistic opportunity for dialogue than the earlier versions of the simulation did. However, it was found that students who had used traditional CAI programs did not readily exploit the power of the natural language system without training in its use.

ECLIPS

Molarity is an area of chemistry where students are known to have difficulty (Gabel, & Sherwood, 1983). ECLIPS is a program designed to understand and solve chemistry problems of molarity typed in plain English. It uses both the natural language system and the generic problem-solver from ALBERT. The result is a program that closely resembles the original ALBERT. ECLIPS can understand and solve problems such as the one shown in Figure 7. Figure 7 also shows what ECLIPS "knows" about the problem and how ECLIPS solved it. At this time, ECLIPS does not include a tutorial component.

The vocabulary for ECLIPS was assembled by analyzing molarity problems found in a collection of introductory chemistry textbooks. The sentences used in these problems were examined and it was found that most of the syntactic patterns were identical to those required for ALBERT. The semantic routines from ALBERT needed some modification but, due to the inherent similarities in the way that information is presented in kinematics and in these problems, the changes were minor. One additional semantic routine was needed to identify chemical formulae and to extract the information from them. ALBERT's problem-solver is

Problem: How many moles of H_2SO_4 are required to make a 0.1 M solution with a volume of 375 ml?

What ECLIPS knows about the problem:

From the problem statement we extract and calculate the following:
We are considering a solution of H_2SO_4
The solute is H_2SO_4 -- atomic mass = 98.08 a.m.u.
The volume of the solution is 0.375 liters
The mass of a mole is 98.07 g
We are asked to calculate the number of moles.
The answer is n = 0.037 mol.

ECLIPS's solution:

We use c = n / V
We rearrange it to give n = c V
Hence n = 0.037 mol.
This is the answer required.

Press RETURN.

Figure 7. An example of a problem solved by ECLIPS.

generic and, in order to make it solve these problems, it was only necessary to substitute the molarity equations for the kinematical equations and change some of the variable names. Chronologically, ECLIPS was the first program derived from ALBERT. A tutorial component was not developed for ECLIPS, as the primary goal of this work was to demonstrate the transportability of the language system and the problem-solver.

CONCLUSION

In this paper we have described four programs that use a natural language system originally developed to teach problem-solving in physics. The success of the natural language processor in each case is due to the fact that the domain of operation is limited. We do not make the claim that this language system will work as well in every situation as it does these programs. However, the scope of many CAI tutorials and intelligent tutoring systems is sufficiently restricted to make ALBERT's language system both useful and robust. The programs described here demonstrate the transportability of the natural language system across diverse subject areas, programming languages and hardware platforms.

The original development of ALBERT required considerable design and programming effort and the work spanned several years. By comparison, the transfer of ALBERT's language system to other domains has required very little programming expertise and has been achieved quite quickly. Some tuning of the language system is required when it is first used in a new subject area. This takes the form of minor adjustments to the vocabulary elements and the syntactic pattern database. However, it has been our experience that this process converges quickly and that the language system soon functions very reliably.

New programs that use ALBERT's language system are under development. The emphasis in these programs is on engaging students in a natural and convincing dialogue to develop concepts that are difficult to learn in the traditional lecture environment. It is our hope that the work reported here will encourage others to enhance their CAI programs with more flexible language interfaces.

References

Brown, J.S., Burton, R.R. & de Kleer, J. (1982). Pedagogical, natural language and knowledge engineering techniques in Sophie I, II and III. In Sleeman, D. and Brown, J.S., editors, *Intelligent Tutoring Systems*. London: Academic Press, 227-282.

Chi, M.T.H., Feltovich, J.P. & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5 (2), 121-152.

Gabel, D.L. & Sherwood, R.D. (1983). Facilitating Problem Solving in High School Chemistry. *Journal of Research in Science Teaching*, 20 (2), 163-177.

Lambiris, M.A.K. & Oberem, G.E. (1993). Natural Language Techniques in Computer-Assisted Legal Instruction: A Comparison of Alternative Approaches. *Journal of Legal Education*, 43(1), 60-78.

McDermott, L.C. (1984). Research on conceptual understanding in physics. *Physics Today*, 37 (7), 24-32.

McDermott, L.C. (1990). Research and computer based instruction: opportunity for interaction. *American Journal of Physics*, 58 (5), 452-462.

McDermott, L.C. (1991). What we teach and what is learned - Closing the gap. *American Journal of Physics*, 59 (4), 301-315.

Mestre, J.P., Gerace, W.J., Hardiman, P.T. & Lochhead, J. (1987). Computer-based Methods for Promoting Thinking in Physics and Algebra: Incorporating Cognitive Research Findings into Software Design. A paper presented at the *Third International Conference on Thinking*, Honolulu, Hawaii.

Novak, G.S. & Bulko, W.C. (1993). Diagrams and Text as Computer Input. *Journal of Visual Languages and Computing*, 4 (2), 161-175.

Oberem, G.E. (1987). ALBERT: A Physics problem-solving monitor and coach. Proceedings of the *First International Conference on Computer Assisted Learning, ICCAL '87*, Calgary Alberta, Canada, 179-184.

Oberem, G.E. & Mostert, O. (1990). Adapting a physics-based natural language processing system for use with English second language students in a diagnostic dialogue setting. A paper presented at the *Tenth Annual Conference of the South African Applied Linguistics Association*, Rhodes University, Grahamstown.

Oberem, G.E., Mayer, O. & Makedon, F. (1992). ILONA: an advanced CAI tutorial system for the fundamentals of logic. *Education and Computing*, 8 (3), 245-254.

Oberem, G.E., Shaffer, P.S. & McDermott, L.C. (1993). Using a Computer to Investigate and Address Student Difficulties in Drawing Free-Body Diagrams. A paper presented at the *Summer Meeting of the American Association of Physics Teachers*, Boise, ID.

Reif, F. (1981). Teaching problem solving: a scientific approach. *The Physics Teacher*, 19 (5), 310-316.

Suppes, P. (1990) Uses of artificial intelligence in computer-based instruction. In Marik, V., Stepankova, O. & Zdrahal, Z, editors, *Artificial Intelligence in Higher Education*. New York: Springer-Verlag, 206-224.

Trowbridge, D. & Chioccariello, A. (1985). Computer dialogues for developing concepts of physics. In Duncan, K. and Harris, D., editors, *Computers in Education*. North-Holland Publishing, 381-385.

Acknowledgments

This work was sponsored in part by Rhodes University and the Institute for Research and Development in South Africa, the Computer Learning Research Center of the University of Texas at Dallas, and the National Science Foundation grant # MDR-8950322 to the Physics Education Group at the University of Washington. Mainframe ALBERT was developed using the TUTOR language of the Control Data PLATO system. PC-ALBERT, FREEBODY and ECLIPS were developed using cT, a product of Carnegie Mellon University. The TenCORE authoring language by the Computer Teaching Corporation was used for CICERO.

An Investigation of the Use of Telecommunications to Increase Equity and Access to Education in Rural Schools in Western Australia

R. OLIVER
Department of Library and Information Science
Edith Cowan University, Bradford St, Mt Lawley, 6050, Western Australia.
E-Mail: r.oliver@cowan.edu.au

T. REEVES
Department of Instructional Technology
College of Education, The University of Georgia, 607 Aderhold Hall, Athens, GA 30602-7144, USA
E-Mail: treeves@moe.coe.uga.edu

Abstract: The provision of diverse and specialist educational programmes to students in rural schools is hindered by many factors associated with the demography and sociology of the schools. This paper reports on a project in Western Australia that used audiographic systems in an attempt to enhance the equity and access to schooling for students in rural schools. Schools collaborated within clusters and used telecommunications systems to create virtual classrooms within which priority educational programmes were delivered. The success of the project and associated factors are discussed in this paper.

Introduction

Western Australia is the largest of the Australian states with a total area in excess of 2,500,000 square kilometres. Ninety percent of the population live in and around the capital city Perth in the South-West. The remaining ten percent of the population live throughout the rest of the state and this demography poses special problems for the delivery of education. In the larger regional centres, there are usually educational opportunities for all children to study at school to the Year 12 university entrance level, however there are frequently restrictions in subject choice that students can make. In smaller regional centres, education opportunities will range from schooling to the Year 10 level, the compulsory schooling age, through to no formal schooling at all. A comprehensive distance education programme is run in the state to cater for students who do not have access to formal schooling. Students can study in a correspondence mode in any grade from pre-school to Year 12.

There are a number of disadvantages apparent in the educational opportunities received by students in remote and rural schools in this state. There is a reduced retention rate among rural students staying on at school to complete Year 12. Secondary school students in smaller country schools are disadvantaged in the subject choices that they can make. The bulk of the beginning teachers (67%) in the state take up their initial teaching positions in country schools. In an attempt to improve the disadvantages of equity and access suffered by rural students, the Ministry of Education in Western Australia has undertaken a number of projects through its Social Justice branch using innovative applications of information technologies. This paper reports on an evaluation undertaken by the authors to assess the outcomes from such a project that sought to use *Telematics* to improve rural secondary students' equity and access to schooling.

Telematics

Telematics is a generic term that describes real-time electronic communications. In this project, the telematics system that was employed involved an audiographic link between teachers and students. Using the software package, Electronic Classroom (Crago, 1992), teaching and learning is achieved through a telecommunications link between computers and an audioconferencing medium using telephone communications. Standard telephone connections are used to connect the teacher and students with two-way voice and graphic communications. Both the teacher and students view the same information on their computer screens which are used as the blackboard while the two-way audio communication is used for the normal student-teacher interaction. A typical setting sees a teacher connected to several remote sites and the establishment of a simulated face-to-face environment through the technology.

The development of Telematics technology in recent years has considerably increased the educational prospects of students in local rural schools. This technology provides the means to deliver interactive instructional programmes across a number of remote and physically isolated sites. It appears to be an ideal medium for the delivery of the specialist programmes. Although this technology can provide such a service, it remains to be seen whether the application will achieve its full potential and provide a real solution to a pressing and serious problem.

Interactivity and Independence. One of the principal strengths of Telematics as a delivery platform for educational programmes in rural schools, is its capacity to mimic conventional face-to-face teaching. Traditionally, the two attributes of face-to-face teaching that are absent from distance education are interactivity and independence (Juler, 1990). All classroom teaching and learning is based on a degree of interactivity between teacher and student. The teacher plays a pivotal role in not only providing instruction, but also in motivating, leading and guiding students. At the same time, schools provide an organised and rigid framework for the learning programme. While schools can therefore be characterised by high levels of interaction and low levels of learner independence, in differing forms of distance education, interactivity and independence tend to be traded off each against the other. Figure 1 shows how these entities compare and contrast across the different modes of distance education (Garrison, 1985).

In many distance education modes and open learning settings, high levels of both interactivity and independence are sought after (Haughey, 1991). These can be achieved for example, through the use of pre-prepared learning materials using alternative forms of learner interaction, including computer based learning materials. For the provision of specialist distance education programmes in a school setting, the preferred blend would appear to involve a high level of interactivity with lower levels of independence. This blend is a close match to traditional classroom teaching and is evident in such delivery systems as interactive television and telelearning courses.

Telelearning. Telematics is a form of telelearning. Telelearning describes the technology that employs telecommunicationand computer mediated communications to create links between teachers and students (Nipper, 1989). Goldman & Newman (1992) describe a number of learning advantages to be achieved with this medium. Telelearning includes such applications as e-mail, audioconferencing and teleconferencing. These learning environments are characterised by active learning situations with student-initiated discourse using interactive communications. Research into the use of telecommunications and in particular, computer-mediated communications, frequently realise significant learning outcomes (Levin, Waugh, Chung & Miyake, 1992).

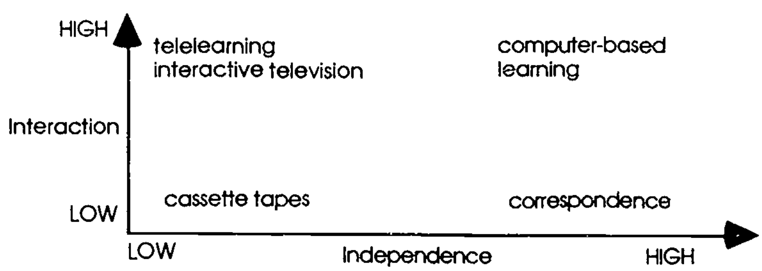


Figure 1
Interactivity and Independence across distance education delivery methods.

Telematics seems ideally suited to delivering specialist programmes into rural schools. Its form is an extension of audioconferencing, a powerful and popular delivery mode of the past twenty years. Audioconferencing provides an auditory link between the teacher and students by which information can be shared, critically analysed and applied in order to become knowledge in the mind of the learner. Audioconferencing is a cheap and effective delivery mode with proven advantages derived from the interaction that it enforces and supports (Garrison, 1990). Telematics or audiographics, as it is commonly called, adds a visual link to strengthen and improve the quality of the audiconferencing interaction.

The Effectiveness of Telematics

There are a number of strategies that have been created as a means to judge the potential of telelearning courses that can be applied against Telematics to test its robustness and integrity as an alternative delivery platform (Stubbs & Burnham, 1990; Collis, 1993). The models that are suggested often differ in the contexts in which they are applied. Collis (1993) describes a strategy that reviews particular applications while Stubbs and Burnham (1990) describe a means to assess the technological dimensions of the application itself. When the criteria described by these strategies are applied to Telematics, few weaknesses appear to exist. Telematics encompasses a wide variety of communication paths, audio, visual and documentary. After training, the system is not judged by its users to be difficult to use. The technology provides a learning environment that approaches the realism of face-to-face better than most others and communications and interactions are instantaneous. It does not offer independence of time and place and thus may be unsuited to other forms of distance education. But as mentioned previously, this is not a requirement of the school based distance education programmes in which it is used, and as such is no impediment to its effective application.

The relative youth of Telematics as an instructional delivery system sees only low levels of previous research and investigation being reported. Audiographics technology has received very positive reviews from studies in the North American context. Positive findings have been reported from early studies in Canada and the USA (Frederickson, 1990). Identified outcomes have included strong attitudinal and motivational gains supporting enhanced learning environments. Several reports in the Australian context have confirmed the potential gains of the technology (eg. States, 1992; Rehn, 1992; McGregor, 1992). In assessing the value of technological applications to teaching and learning, we must be careful to avoid the "visionary" trap described by Collis (1993). This trap is encountered when one assumes because a project proceeds and happens as planned, this can be used as a measure of its success. Measures of success need to be empirically based and empirical evaluations of Telematics are only just happening.

Evans and Nation (1992) describe an empirical investigation into Telematics in Victorian schools. Their preliminary report suggests the need for some caution in embracing the technology as evidenced from some shortcomings and impediments identified in their research. Some of the factors that can limit the effectiveness of the teaching and learning environments are:

- the very high levels of preparation and planning that necessarily must accompany lesson delivery,
- the problems associated with establishing discipline and authority in the virtual classroom,
- technical problems that can arise, and
- the changes brought about to work practices.

A study of the use of Telematics among student-teachers (Stacey, 1994) describes some very positive outcomes achieved in a teacher education programme. Through the use of Telematics in a LOTE practicum environment (the teaching of languages other than English), the researchers identified a number of positive gains for both teachers and students through Telematics interactions. The researchers report such findings as;

- increased levels of student participation in lessons,
- increased use of oral language by teachers and students,
- the improvement of children's listening skills, and
- more flexible and cooperative learning environments.

Telematics has made significant in-roads in Australian distance education programmes in the past three years. The technology has proved itself to be robust and well able to support the educational applications in which it is used. While previous research has demonstrated the utility and efficacy of the technology, there is a need for research to further investigate factors influencing the effectiveness of the teaching and instructional applications.

The PCAP Project

The study that is described in this paper was undertaken as an evaluation of a large scale implementation of Telematics into rural schools in Western Australia, called the PCAP (Priority Country Areas Programme) Project. The stated aims of the project were to:

- increase the number of students in prescribed country areas completing a least twelve years of schooling,
- expand opportunities available to these rural school leavers,
- improve levels of student achievement,
- reduce incidence of gender bias in subject choice (particularly among students from low socio-economic backgrounds,

- increase opportunities for personal development through increased interaction with peers from broader cultural and socio-economic backgrounds.

Special funding was supplied by the Federal government to give states the opportunity to explore ways to improve equity and access to education in rural regions. Four regions in Western Australia were chosen to participate in the project after a submission process had been completed. These regions extended from the far North-West to the Southern regions. Within each region clusters of schools cooperated and collaborated to deliver educational programmes among themselves using Telematics.

The special funding provided all schools in the cluster with the required equipment to run Telematics courses and provided training and technical support to teachers in the rural schools. In the first phase of the programme. funding was provided to enable teachers to be released from normal duties to deliver Telematics programmes to other schools. It was planned that once schools and teachers had become familiar with the technology and its application, they would become self-sufficient and able to use the technology in subsequent applications without further financial support. Another purpose of this study was to determine whether the objective of self-sufficiency could be obtained and whether Telematics had the potential to significantly enhance the equity and access of rural schools to educational programmes.

The researchers used a multi-method technique of data gathering and analysis developed and applied in similar projects in the USA (Reeves, 1992). This technique was based on the work of Mark and Shotland (1987). The study used qualitative data gained from multiple sources including observation, interview, questionnaires and documentation and multiple perspectives, to enable validity and reliability to be maintained. The data gathering and evaluative process was built around the four stages of the process described by Collis (1993);

- gathering information about the intentions of the project,
- assessing logical contingencies,
- observing and measuring actual and contextual applications,
- assessment of incongruities between intended and actual occurrences.

The researchers travelled widely to visit the participating schools to discuss aspects of the project with the teachers, administrators and students. The teaching and learning processes were extensively observed to enable judgements to be made about the success of the project and factors that influenced the successes achieved.

Outcomes

The main purpose of the PCAP project was to provide an enhanced and broadened curriculum within rural schools that would contribute to post-compulsory (P/C) schooling programmes. In the Kimberley region, schools chose to use Telematics to deliver post-compulsory programmes while in the others, the schools implemented programmes among younger children that would lead to increased opportunities when they moved into the post-compulsory phase of their education. The curriculum areas that formed the PCAP project are shown in Table 1.

Table 1
The Curriculum Choices within the Regions

Region	Curriculum	Technology
Kimberley	Applied Computing (P/C) English as a Second Language (P/C)	Telematics
Geraldton	Maths, English, Social Studies Unit Curriculum	Telematics
Pilbara	Languages other than English Japanese	Telematics
Kalgoorlie	Languages for students at risk, English, Transition Mathematics	Telematics

A typical programme would see a selected teacher in one school prepare a teaching programme and curriculum materials for students enrolled in other schools in the region. Selected students would have some element of their school programme replaced by the Telematics programme. The lessons would be timetabled several times per week and conducted in the Telematics room through a system of audioconferencing and computer communications. The educational programmes prepared by the Telematics teachers would include activities for students to complete outside class time together with assignment and project material. Students would work on these activities between lessons to prepare for further lessons and to practice and consolidate the content of previous lessons.

Telematics teaching was judged to have been very successful by a large majority of the participating schools, teachers and students. The delivery method worked well and was seen to be an effective way to deliver the specialist programmes into the rural schools;

- students enjoyed the format of the electronic lessons and looked forward to them,
- the interactivity between teachers and students and students and students enhanced the quality of the learning outcomes,
- students assumed high levels of responsibility for themselves and their work,
- the system enabled specialist programmes to be delivered to schools without specialist teachers.

There were varying levels of success achieved among the four projects and a number of key factors were observed to which the degrees of success could be attributed. The major influences came from the ways by which teachers used the technology in their teaching and learning and the manner in which the Telematics programmes were organised within and across the schools.

Instructional Strategies. Telematics is a technology that demands high levels of technical expertise from its users as well as the use of appropriate instructional strategies. All teachers in this project were experienced classroom teachers who had learned to use Telematics over a relatively short period of time. There were naturally many differences observed in the nature and format of the instructional programmes that each delivered. From observations and interviews, the researchers judged that the following aspects of the teaching and learning processes were significant factors in the successes (or lack thereof) achieved by the teachers;

- Levels of preparation and planning accompanying lesson delivery, including use of varying instructional formats and strategies, appropriate levels of student-initiated and independent-student activity, and discerning use of the technology matched to intended learning outcomes.
- The development of flexible learning environments to promote positive teacher-student relationships, open and comfortable classroom climates promoting student involvement and interaction. Activities that created and facilitated contact between the teachers and students to further develop relationships and interpersonal communications.
- Effective communications skills on the part of teachers enabling appropriate mixes of teacher-led and student-led communications to be supported.
- Strong administrative and teaching support at the receival school including established and consistent organisational systems.
- Strong technical skills on the part of teachers (and students).
- Constant evaluation and assessment on the part of the teachers on all aspects of the teaching and learning.

Organisation and Management. The success of the teaching was seen to depend not only on the quality of the lessons delivered by teachers but also on a myriad of other activities and supports influenced by the ways in which schools implemented and managed their Telematics programmes. The researchers identified factors at the school, regional and system level that impacted significantly on programme success. These included:

- At the *school level*, the setup and physical location of the equipment within schools, the level of supervision and coordination provided by teachers at the receival schools for the delivery programmes, and the extent and depth of the skills and expertise in Telematics across the curriculum areas and among staff.
- Within *regions*, the level of cooperation and collaboration in the planning and management of Telematics programmes undertaken within local clusters or groups of schools, the adoption of agreed policies and practices relating to programme delivery and receival, and the capacity of schools to exchange programmes and courses of instruction.
- At the *system level*, the provision of training and support programmes provided to support Telematics, the level of on-going research and development seeking ways and means to improve teaching and learning and the overall levels of computing skills and technology awareness among teaching graduates and teachers within rural schools.

A key indicator of the success of the project was the extent to which the schools acted in collaboration and cooperation to ensure the continuation of the programmes and to extend curriculum offerings and applications. Across all schools there was recognition of the strong need for educational programmes that enhanced the prospects of their students in post-compulsory schooling. Telematics was judged to be successful, cost-effective and capable of delivering these programmes. By the end of the first year of the project, schools were required to fund their own programmes and innovative schemes and strategies were employed to achieve this;

- Larger schools with more flexibility in programme selection and organisation delivered concurrent programmes to face-to-face students and classes of students in remote schools,
- connections were made beyond regions to enable students within schools to join other Telematics programmes,
- collaborative programmes between schools were continued and resourced from school funds,
- school timetables and staffing arrangements across regions were reorganised to fit Telematics requirements.

Conclusions

In the local context, Telematics showed itself to be an ideal application of technology in response to an urgent educational problem. The technology itself was generally quite robust and reliable. As an instructional delivery platform, it is flexible and efficient and provides a number of key elements associated with effective teaching and learning. It demands reasonable levels of technical knowledge and expertise of the part of the users. The application is in its infancy and will obviously benefit from research and development activity into effective applications. From this study, it is evident that there are many factors that can influence the successful delivery of programmes through Telematics. Applications of the findings across local programmes will help to enhance the utility and efficacy of as an alternative delivery platform for rural school educational programmes. A key area for further research is the development of instructional strategies for the different curriculum areas to enable Telematics teaching to create learning environments that increase the students' level of participation as active and reflective learners. Our university is including Telematics as a component of its pre-service teacher education programme and is looking to establish a research direction in this field.

References

Collis, B. (1993). Evaluating instructional applications of telecommunications in distance education. *Educational and Training Technology International, 30*(3), 266-274.

Crago, R. (1992). *Electronic Classroom 2.0.* Brisbane: Revelation Computing.

Evans, T., & Nation, D. (1992). Teachers' experiences of teaching by telematics: some preliminary research findings. In *Australian Association for Educational Research,*, Geelong: Deakin University.

Frederickson, S. (1990). *Audiographics for distance education: An alternative technology.* Paper presented at the Annual Conference of the Alaska Association for Computers in Education, March, 1990.

Garrison, D. (1985). Three generations of technological innovations in distance education. *Distance Education,* 6(2), 235-241.

Garrison, D. R. (1990). An analysis and evaluation of audio teleconferencing to facilitate education at a distance. *The American Journal of Distance Education, 4*(3), 13-24.

Goldman, S., & Newman, D. (1992). Electronic interactions: how teachers and students organise schooling over the wires. *Interactive Learning Environments, 2*(1), 31-44.

Haughey, M. (1991). Confronting the pedagogical issues. *Open Learning, 6*(3), 14-23.

Juler, P. (1990). Promoting interaction: maintaining independence: swallowing the mixture. *Open Learning, 5*(2), 24-33.

Levin, J., Waugh, M., Chung, H., & Miyake, N. (1992). Activity cycles in educational electronic networks. *Interactive Learning Environments, 2*(1), 3-13.

Mark, M., & Shotland, R. (Ed). (1987). *Multiple methods in program evaluation.* San Francisco, CA: Jossey-Bass.

McGregor, A. (1992). *Networks on trial: An evaluation of the Western Australian Learning Networks Centres Project.* Western Australian Office of Higher Education, Perth.

Nipper, S. (1989). Third generation distance learning and computer conferencing. In R. Mason & A. Kaye (Eds.), *Mindweave: Communication, Computers and Distance Education* Oxford: Pergamon.

Reeves, T. (1992). Evaluating schools infused with technology. *Education and Urban Society Journal,* 24(4), 519-534.

Rehn, G. (1992). *The Western Australian networks for learning trial: Overcoming the problems of distance.* Proceedings of the Australian Society for Educational Technology Conference, Adelaide.

Stacey, E. (1994). Technology overcomes Australian distances. *Technological Horizons in Education, 21*(6), 56-59.

States, I. (1992). Interactive teaching/learning and distance education. In J. Herrington (Eds.), *Distance Education: Future Visions* Perth: WADEC.

Stubbs, S. T., & Burnham, B. R. (1990). An instrument for evaluating the potential effectiveness of electronic distance education systems. *The American Journal of Distance Education,* 4(3), 25-36.

Hypermedia Representations for Learning: Formal and Informal Observations on Designs and Directions

A. P. PARKES
Computing Department
Lancaster University, Lancaster, LA1 4YR, England
E-Mail: app@uk.ac.lancs.comp

Abstract: This paper focuses on issues of representation and interaction styles that are relevant considerations in the design and analysis of hypermedia learning environments. The *representational* issues discussed are: *analogical representations vs. fregeian representations, multiple representations vs. single representations* and *replete representations vs. minimal representations*. In terms of *interaction styles* found in hypermedia, we discuss *constrained vs. unconstrained* and *exploratory vs. constructive* approaches. Supporting examples from studies carried out to investigate some of the issues discussed are used by way of illustration. The paper concludes by discussing the cultural influences that are impeding the effective use of non textual representations as primary communication media.

The purpose of this paper is to outline several concepts and ideas relating to the analysis and design of learning environments, particularly those of the hypermedia variety. The paper is not meant to provide answers to the questions posed by the enterprise of hypermedia interface design, but rather to discuss certain perspectives on the questions themselves. Thus, the paper is intended to provoke further discussion about issues relating to hypermedia interface design and evaluation. While our main interest is in using predominantly graphical hypermedia representations in the learning of mathematics and other formal domains, the paper discusses issues that we believe are demonstrated to be central to hypermedia interface design.

The structure of the paper is as follows. The next section analyses some design choices referring to the nature of the representations that we use in our hypermedia interfaces. The following section then discusses similar binary differentials with respect to the style of user activities demanded by hypermedia representations. (Please note that neither section is claimed to represent an exhaustive list of topics). The analysis is supported by examples from the "HUGH" series of studies involving learners in reasoning about formal hypermedia representations (Parkes 1992a; 1992b; 1994). The HUGH systems embody constrained direct manipulation graphical and text representations of theorem proving tasks in formal computer science. Examples from the HUGH studies (one HUGH screen is shown in Figure 1) are used to illustrate the issues raised in the paper.

Design Choices for Hypermedia Representations

Analogical Representations vs. Fregeian Representations

The terms "analogical" and "fregeian" were used a few years ago by a leading AI theoretician, A. Sloman (Sloman, 1975; see also Hayes, 1974), long before it became fashionable to talk of "media" (let alone "multimedia" or "hypermedia") in the context of computing. Sloman's fairly abstract discussion concerns concepts at the heart of the whole business of representation. Analogical representations are those in which relationships in the represented entity are symbolised by *actual relations* (as opposed to names of relations) in the representation of that entity. A map is an obvious example of this, since spatial relationships between objects on the ground are reflected in the map (unless it's a map of the London Underground system).

Sloman (*op cit.*) also describes how mathematics teaching neglects analogical representations i.e. pictures and diagrams. We observe that a central problem with respect to analogical representations in mathematics is in choosing or designing a representation (e.g. suitable diagram, picture, etc.) that is really useful to learners. In a formal domain, the picture can become as abstract as the usual text-based notation, and in some cases it appears that one has already to understand what the diagrams refer to before the diagrams can be understood! For example, the initial HUGH studies were devoted to investigating the understanding that learners have of a direct manipulation diagrammatic representation of an abstract mathematical problem solution. The diagram was similar to that on right of Figure 1, and a number of simple operations could be carried out by the learner, such as copying significant parts of the diagram into a "store" (lower left of Figure 1), then replacing significant parts of the diagram by the component in the store. Sequences of operations that the learner performed were constrained to be meaningful steps in the proof of a theorem. The main aim of the studies was to find out if the learners themselves could arrive at a description of what these "steps" were signifying. As was demonstrated by these studies, users can happily directly manipulate abstract diagrams without arriving at the account of their semantics that the designer intended. Viewed purely syntactically, the operations were very simple, while their meaning was complex, and it required substantial mental effort for the learners to construct it.

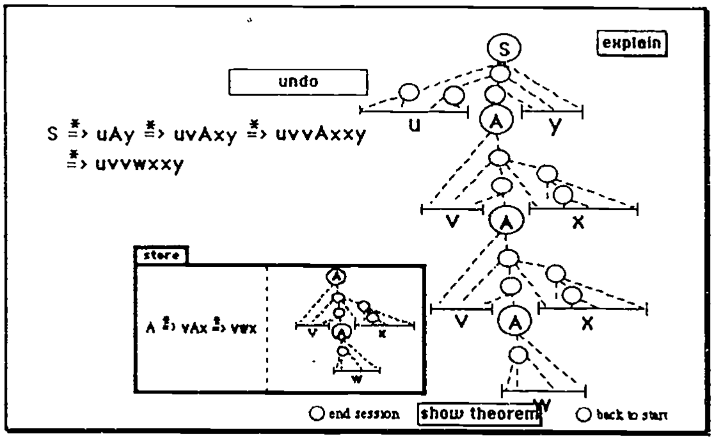


Figure 1. A screen from the dual representation version of HUGH

Multiple Representations vs. Single Representations

A critical design issue is the extent to which we provide the user with one representation or several, or more specifically, the extent to which we use one encoding expressed in one medium or several encodings expressed in a variety of media. We need to be clear about why we might want to use several representations, because there are clearly different reasons for choosing to do so. For example, it may be so that everyone has a chance to understand the material (c.f. Draper's (1992) search for representations that work for all, or diSessa's (1986) call for "the equivalent of pens and paper for everyone" [p126]). It may be to provide variety for a single user, or to ensure redundancy, or to reinforce. It may be to provide access to features that one medium cannot express, or cannot express to the same degree of veracity, as another medium.

Moreover, multiple representations should exert an influence over the ways in which we attempt to assess the knowledge gained by learners. Implicit in a system that presents multiple representations is an assumption that the learner is able to reach an understanding of the relationships between the various representations. It has been argued (White, Sandberg, Behar, Mockler, Perez, Pollack & Rosenblad, 1984) that asking a learner to translate from one medium to another requires a symbolic transformation which may distort his or her comprehension. White *et al.* also argue (c.f. McLuhan) that the medium affects the message, i.e. influences the coding *processes* that the receiver of the message can apply to the message. It has been demonstrated that, for child learners at least, the medium of presentation affects what is learned and how it is learned, while the medium of *comprehension testing* may also affect the evaluation of what knowledge is acquired if it requires subjects to transform information received in one medium into another in order to respond (White and Pollack, 1985). White and Pollack's study suggests that having to perform such translations when demonstrating comprehension can decrease comprehension scores. Tacit support for this view resulted from the HUGH studies, when learners were asked to verbalise their thoughts on the meaning of the operations. Users of a version with diagrams alone (i.e. a version without the textual representation beneath the "undo" button, and on the left of the

"Store" in Figure 1) gave explanations initially expressed purely syntactically, e.g. "the diagram in the store replaces the selected part of the diagram", that only later evolved into more semantically-oriented descriptions. On the other hand, users of a dual representation version (a display from which is shown in Figure 1), gave semantic descriptions much earlier on in the session, thus suggesting that the textual representation was encouraging them to translate between the representations, the text providing links to facilitate the verbal description of the semantics of the operations. It is noteworthy that this occurred *despite* the fact that learners using the dual representation version overwhelmingly preferred to interact with the diagram - the textual representation was exerting a subtle influence that was not apparent from the learners' actions in the interface.

As yet, we lack knowledge about what are the optimal combinations of representations (and corresponding media) for given tasks and learners. Beasley and Vila (1992) suggest that future research on media should be conducted to study the attributes of media with respect to the cognitive implications of those attributes. Larkin and Simon (1987) provide what might prove to be a *theoretical* basis for assessing the cognitive implications of various media. They essentially say that we need to be more aware of the ways in which a "chunk" of information can be presented by different media, rather than focusing on the information content *per se*. One useful contribution that Larkin and Simon make is a richer model of *equivalence*. They distinguish between two types of representational equivalence: equivalence in the information *content* of different representations (*informational equivalence*), and equivalence in the information *processing* requirements (for the perceiver) of the representations (*computational equivalence*). Two representations can sometimes be informationally equivalent but differ considerably in the information processing demands they make upon the user.

The issue of the "computational cost" (to the learner) of translation between media emerges in the differences between the explanations of the users of the "diagram" *vs.* "diagram plus text" versions of HUGH. Users' misconceptions became apparent to us only when they attempted to *describe* their understanding of the representations. The dual representation users seemed to benefit from the aid given in the translation process by the presence of the textual representation, and the ways in which its own behaviour modelled that of the favoured representation for interaction (the diagram).

Replete Representations vs. Minimal Representations

The author recently attended a conference at the Glushkov Institute for Cybernetics which is in a residential area of Kiev. On the first morning of the conference he walked from his lodgings along a main road to the conference site. He knew that the Institute was somewhere along this road, but was also aware that it was set back a few hundred yards from the road, behind other buildings, and he could quite easily pass it without realising. The author possessed a plan of the city on which neither the scale of the map nor the location of the institute were indicated. None of the streets seemed to have name plates. The author approached an old man standing by a bus stop, and used a combination of pointing (both at himself and the portion of the map representing his approximate location) to convey in a few moments the question "where am I (on this map)?". The old man spoke to the author in Ukrainian, while the author replied in English. Suddenly the old man spoke two words that the author recognised: "Glyooshkova", and "Keebernetikka" (the old man must have guessed that there was just no other reason for a foreigner to be in that part of town). The problem was solved: in a few more gestures the old man had shown the author how to get to the institute!

The striking feature of this episode was the way communication was effected using a minimal level of representation. There was no need to show the old man a video of the institute, or even a text description. Questions were asked by pointing at part of a map - the rest was done by simple gestures that seem to be more or less universal.

The above episode is relevant to multimedia and hypermedia interface design. We are often tempted to make a system "replete with different representations" (or to provide a single "replete representation"). As a reaction to this, we have experimented with systems that simplify, or *minimalise* the representations, and attempted to investigate what can be achieved by the simplest possible representation. Replete representations may simply overload the user, i.e. Aaronson and Carroll (1987) may be correct in asserting that learners are very situational and often find information useless if they cannot relate it to the solution of their problem.

In the HUGH studies users explored and described graphical representations that were supported by very little explanatory text. This is based on our assumption that before we start *adding* media such as diagrams to systems, we need to know what we can, and cannot, hope to achieve by doing so. In this we were agreeing with Draper (1992, [p177]), who says, "adding media...[to a system]...is of no automatic value - it will only be useful if someone invents a representation that fits the mind in some important way and happens to use the extra 'media'". He also goes on to say "Representations...[for]...promoting understanding are usually simplified, not enriched in complexity" [p178]. It is interesting to compare the latter assertion with the claims of the constructivists, some of whom seem to suggest that complexity may be a desirable feature (Jonassen, Ambrusso, & Olesen, 1992). Whether this is true or not, it is not useful as a design aim, because we cannot evaluate a "complex" representation against a "simple" one and say that any gains (however measured) are due to the complexity: we just wouldn't know if it was the complexity itself that caused the gain. This is especially true in terms of multiple representations, where there are so many poorly understood dimensions to complexity (i.e. the complexity of any one encoding, the complexity of the mental mapping between different simultaneously presented encodings...). In a way, our approach is not the typical one of presenting all the information to the learner, and then establishing that something is being learned. We actually say: *present the learner with the minimal amount of information, and see from this what they cannot learn; then investigate what form of information we need to add to alleviate the problem.* The initial HUGH system (a purely diagrammatic version) featured a simplified form of the tree shown right of Figure 1. We realised that learners were attributing significance to features of the tree such as its *symmetry* (unlike in Figure 1, the tree was symmetrical about a central vertical path), but this was not meant to happen. This inference, in turn, resulted in insufficiently general conclusions being made by the learners. Thus, we were able to decide firstly, that the diagram needed to be redesigned, and secondly, what sort of supporting information needs to be given to ensure that the noted problem is overcome. The minimal representation approach was a major facilitator of the accuracy with which we could identify the source of the problems that our representation presented to learners.

Styles of User Activity in Hypermedia Representations

Constrained Activity vs. Unconstrained Activity

Much used to be made of the freedom granted to the user in a hypermedia system. Indeed Mülhäuser (1992) compares the "anarchy" of current styles of computer assisted instruction (CAI) with the "dictatorship" of old style "presentation-based CAI". However, by definition, any hypermedia system is constrained in some senses, just by virtue of the fact that some things are linked and others are not. The issue here is the extent to which the imposition of additional constraints can be a useful feature. By this we mean more than just facilities such as "guided tours" that are an embellishment to some systems. Rather, we refer to the extent that transitions made by the user between hypermedia states are designed to reflect a more useful, perhaps *procedural* meaning than is usual with hypermedia links. One such design is provided by the Harmony Space system (Holland, 1992), where novices to music can learn complex composition skills by making simple gestures with the mouse in a graphical representation, resulting in "well formed" chord progressions being played on a synthesiser.

In a slightly more trivial way, some of the HUGH systems also represent this approach, by modelling a proof, as we have indicated earlier, as a sequence of "cut" and "paste" operations on a graphical representation. The difference in the HUGH case is that there is not such an obvious relationship between operations and their conceptual outcomes (no music to the ears of the learners), which the HUGH users have to arrive at themselves.

Exploratory Learning vs. Constructive Learning

A contemporary issue in hypermedia design is whether users learn more from exploring a representation or from building one for themselves. At the forefront of this debate are constructivists (e.g. Cunningham, Duffy, & Knuth, 1993), who see knowledge acquisition as involving the learner in the construction and interpretation of knowledge, as opposed to passive reception or unstructured exploration. The realisation of such a theory in hypermedia learning environments suggests systems that enable learners to build representations. (Though it is interesting to contrast this approach with that proposed by Laurel (1986), who asserts that people want to move around inside a representation, not make one). In a constructivist vein, Hammond (1991; 1992), describes a "knowledge jigsaw" task, where users construct an argument by linking text boxes together, with the links representing reasoning steps in the argument.

The latest HUGH study focuses on users' performance in one of two versions of a theorem proving task. One version is exploratory, and involves carrying out operations of the "cut" and "paste" style mentioned above (a display from this version is shown in Figure 1). A second version is (independently of Hammond's use of the term) called a "jigsaw'. In this version users construct a graphic by dragging the "pieces" around the screen so that they "snap" together (see Figure 2). The interface is again constrained, in this case so that the pieces only actually fit together in meaningful ways. There are significant differences between the two versions in terms of the nature of the task, despite the fact that, while being predominantly either *exploratory* (finding the sequence of actions that lead to the proof) or *constructive* (building up the representation from its component parts), they have several elements in common. In the exploratory version a representation is being built, and in the constructive version exploration is needed to discover the correct assembly. Thus far, there seem to be no noticeable differences in the explanations from users of either version (this differs from the results obtained with the single *vs.* dual representation versions - see above). What *is* different is that the predominantly constructive version seems to result in the learners giving much more thought to the representation, because they have to decide to move a particular piece to a particular destination, and cannot (as in the exploratory version) simply click on various objects to see what the response of the system will be. We are currently designing experiments to see if the extra cognitive effort required by the constructive version results in increased learning, or if it is merely an additional mental overhead which is not present in the exploratory version.

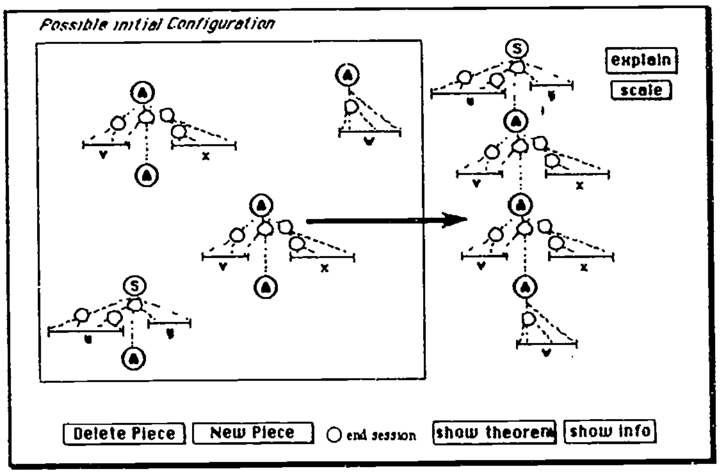


Figure 2. A screen from the HUGH "jigsaw" version (inset shows possible initial state)

Discussion

At a more general level than that of the preceding discussion, the problems associated with the design of external representations in hypermedia learning environments result, at least in part, from our being products of what has been called a "logocentric" educational culture, i.e. a culture that emphasises talk and written language to the near exclusion of any other mode of representation (Cunningham, Duffy, & Knuth, 1993). Kress and van Leeuwen (1990) suggest that our culture is not opposed to visual media *per se*, but to what they call the emerging visual literacy, as it is a threat to the dominance of verbal literacy. This dominance of verbal language has, they say, been a major cause of our inability to fully understand what is actually communicated by images. Although not addressing problems of computer system or interface design, Kress and van Leeuwen actually indicate one important source of the difficulty of hypermedia interface design:

> "Newspapers, magazines, public relations materials, advertisements ...[etc.]...involve a complex interplay of written text, images and other graphic elements...[which]...combine together into visual designs, by means of layout. The skill of producing texts of this kind...is not taught in schools. In terms of this new visual literacy, education produces illiterates" (Kress and van Leeuwen, 1990, [p2]).

The opposition to non verbal literacy takes many forms, and was recently heard in an academic discussion about diagrammatic representations when a participant was arguing fiercely that there was a level of complexity beyond which diagrams could not go, but written language could. This is unfair to diagrams on two counts. Firstly, most learners, whether subjected to verbal *or* diagrammatic information, seem to encounter a level of complexity beyond which they find it difficult to proceed. Secondly, we are not (yet) as a culture as experienced at communicating with diagrams as we are with pictures (as most diagrammatic forms we are exposed to in our education are usually, concrete, simple, and have the status of supporting illustrations). In the future we may teach an understanding of the non-verbal media that will enable us to educate much more effectively than we ever could by the use of words alone.

References

Aaronson, A. & Carroll, J. M. (1987). The answer is in the question: a protocol study of intelligent help. *Behaviour and Information Technology*, 6, (4), 393-402

Beasley, R.E., & Vila J.A. (1992). The identification of navigation patterns in a multimedia environment: a case study. *Journal of Educational Multimedia and Hypermedia*, 1 (2) 209-222.

Cunningham, D.J., Duffy, T.M., & Knuth, R.A. (1993). The textbook of the future. In C. McKnight, A. Dillon & J. Richardson (eds.), *Hypertext: A Psychological Perspective*. Chichester: Ellis Horwood.

diSessa, A.A. (1986). Notes on the future of programming: breaking the utility barrier. In D.A. Norman & S.W. Draper (eds.) *User Centered System Design*. Hillsdale, NJ: Lawrence Erlbaum.

Draper, S.W. (1992). Gloves for the mind. In A.M. Kommers, D.H. Jonassen, & J.T. Mayes (Eds.) *Cognitive Tools For Learning*, Heidelberg: Springer-Verlag.

Hammond, N. (1991). Teaching with hypermedia: problems and prospects. In H. Brown (ed) *Hypermedia/Hypertext and Object-oriented Databases*. London: Chapman and Hall.

Hammond, N. (1992). Tailoring hypertext for the learner. In A.M. Kommers, D.H. Jonassen, & J.T. Mayes (eds.) *Cognitive Tools For Learning*. Heidelberg: Springer-Verlag.

Hayes, P.J. (1974). Problems and non-problems in representation theory. In Proceedings of the *AISB Summer Conference*, University of Sussex, Brighton, 63-79.

Holland, S. (1992). Interface design for empowerment: a case study from music. In M. Blattner & R. Dannenberg (eds.), *Interactive Multimedia Computing*. Reading, MA: Addison-Wesley.

Jonassen, D.H., Ambrusso, D.R., & Olesen, J. (1992). Designing a hypertext on transfusion medicine using cognitive flexibility theory. *Journal of Educational Multimedia and Hypermedia*, 1 (3), 309-322.

Kress, G. & van Leeuwen, T. (1990). *Reading Images*. Victoria, Australia: Deakin University Press.

Larkin, J.H. & Simon, H.A. (1987). Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11, 65-99.

Laurel, B.K. (1986). Interface as mimesis. In D.A. Norman & S.W. Draper (eds.), *User Centered System Design*. Hillsdale, NJ: Lawrence Erlbaum.

Mühlhäuser, M. (1992). Hypermedia and navigation as a basis for authoring/learning environments. *Journal of Educational Multimedia and Hypermedia*, 1 (1), 51-64.

Parkes, A.P. (1992a). The inferential appropriateness of a manipulable inter-medium encoding. In M. Blattner & R. Dannenberg, (eds.), *Interactive Multimedia Computing*, Reading, MA: Addison-Wesley.

Parkes, A.P. (1992b). Hypermedia, representation and inference in the learning of formal computer science. In Proceedings of conference: *Developments in the Teaching of Computing Science*, University of Kent, Canterbury, April 6-8, 1992, 80-87.

Parkes, A.P. (1994). A study of problem solving activities in a hypermedia representation. To appear in *Journal of Educational Multimedia and Hypermedia*.

Sloman, A. (1975). Afterthoughts on analogical representations. In Proceedings of the Conference on *Theoretical Issues in Natural Language Processing*, Cambridge, MA, 1975, 164-168.

White, M.A. & Pollack, J. (1985). Within and across modality comprehension of electronic media in children. *International Journal of Man-Machine Studies*, 22, 209-214.

White, M.A., Sandberg, B., Behar, E., Mockler, J., Perez, E., Pollack, J., & Rosenblad, K. (1984). Preserving the integrity of the medium: a method of measuring visual and auditory comprehension of electronic media. *International Journal of Man-Machine Studies*, 20, 511-517.

Acknowledgements

The HUGH studies were supported to 1990 by the UK Science and Engineering Research Council and to date by the Computing Department, Lancaster University.

Integrating Existing Applications in Hypermedia Learning Material (General Issues & Experiences with OLE Technology)

H.W.J. BORST PAUWELS, J. SOUSA PINTO, B. SOUSA SANTOS, J.A. MARTINS
INESC and Department of Electronics & Telecommunications
University of Aveiro, Aveiro, 3800, Portugal
E-mail: hbp@zeus.ci.ua.pt

Abstract: Integration of existing applications in hypermedia environments is a promising approach towards more flexible and user-friendly hypermedia learning material. In this paper we describe some issues, methods, and observations related to the design of a hypermedia courseware editor that is capable of integrating existing applications.

The aim of distance education environments is to provide learning resources to large numbers of students spread over wide geographical areas. The social significance of these systems is twofold. On one hand, they are helpful to bridge the gap between isolated students and learning centers. On the other hand, they allow teachers to be efficiently 'shared' over several geographically distributed students. Traditionally, learning material in distance learning environments consist of rather static material like printed text, audiocassettes and videocassettes. Computer supported courseware based on hypermedia technology opens new perspectives for distance education. Hypermedia learning material can be developed in less time, is more flexible for modifications than traditional learning material and can be distributed relatively fast over computer networks. Integration of existing applications in hypermedia environments (Nielsen, 1990) is a promising approach towards more flexible and user-friendly hypermedia learning material.

Objectives of the design of a hypermedia courseware editor

Our first objective was to design a hypermedia courseware editor that is capable of integrating existing applications. From the engineering point of view the advantage is obviously. We follow the object oriented approach which basically promotes efficiency by re-using existing software. From the end user's point of view we can mention the advantage that integrating existing applications in a hypermedia environment allows students and teachers to use their familiar set of applications in order to create or edit multimedia objects. As a consequence users will need less time in learning to work with the new environment.

A second objective consisted in obtaining a modeless editor which supports both authoring and reading of hypermedia documents based on one underlying metaphor. With this objective in mind we want to avoid the traditional gap that exists between on one hand a relatively small group of "expert" hypermedia authors and on the other hand a large group of hypermedia consumers.

Methods of the design of a hypermedia courseware editor

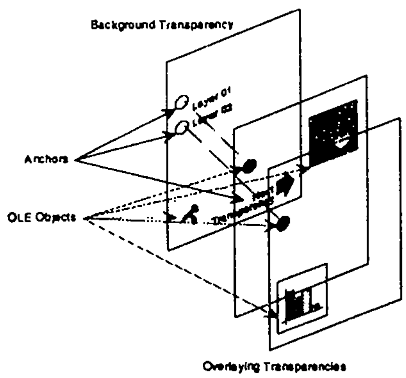
OLE technology

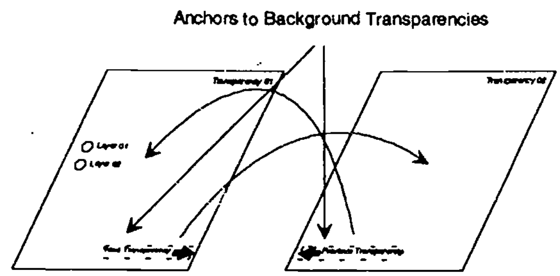
We developed a hypermedia courseware editor called HyDE (**Hy**permedia **D**ocument **E**ditor) (Borst Pauwels, Pinto, Sousa Santos, & Martins, 1993). We integrated existing applications by using Microsoft ® Windows TM OLE technology. OLE, Object Linking and Embedding, stands for an extensible protocol that enables an application to use the services of other applications. Basically, this implies that a user does not have to leave an application (OLE client) in order to create or edit application specific data, but instead more comfortably, simply launch an application (OLE server) by a mouse click on the specific data.

Electronic Overhead Projector

HyDE acts as an OLE client capable of receiving OLE data from several commercial OLE servers. An overhead projector was used as the underlying metaphor. Our editor act as an "electronic" overhead projector on which lessons composed of piles of "electronic" transparencies, can be displayed either in simple sequential order or in more sophisticated so-called "web" structures. More precisely, in our approach, one hypermedia node consists of a background transparency which may, or may not, be covered by one or more overlaying transparencies.

Background transparencies can be filled with OLE objects and afterwards connected by means of anchors, either to other background transparencies or to overlaying transparencies (Figure 1).





Fignre 1. Transparency organization **Figure 2.** Anchors connected to background transparencies

An anchor is an rectangular area on a background transparency. When it is introduced a new blank background transparency will be automatically displayed or, in case the user requested a connection to an overlaying transparency, a new blank overlaying transparency will be automatically inserted over the background transparency. In both cases, the new blank transparency can be filled with OLE objects. A mouse click on an anchor connected to a background transparency will trigger an event that displays again the associated background transparency (Figure 2).

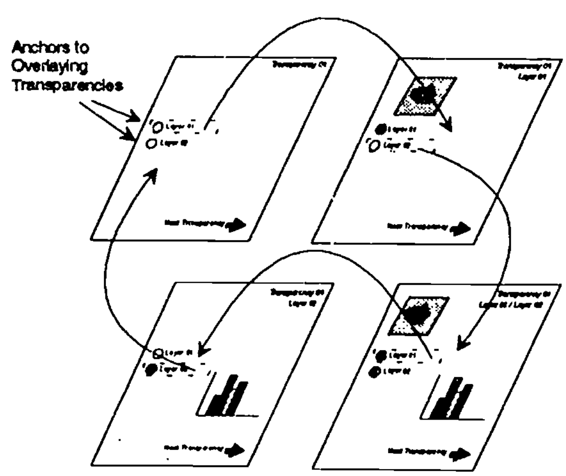


Figure 3. Anchors connected to overlaying transparencies

A mouse click on an anchor connected to an overlaying transparency will trigger an event that inserts the associated overlaying transparency over the background transparency or, in case the overlaying transparency is already inserted, removes it (Figure 3).

For a more detailed description of the functionality of HyDE see (Pinto, Borst Pauwels, Sousa Santos & Martins, 1994).

General issues related to hypermedia learning material

Hypermedia stands for a specific kind of information system in which information is stored in graph- or web- structures. The basic unit of such information systems is a node which can be linked to other nodes by mean of links. This non-linear aspect of storing information is fundamental for both hypertext and hypermedia. They allow users to read or traverse documents in a non-linear fashion. In this section we indicate some issues related to hypermedia learning material in distance education environments.

Document

The role of hypermedia documents in distance learning environments is twofold. Firstly, a hypermedia document may contain a course which can be downloaded on the student's workstations. Subsequently, the teacher gives a remote lecture by presenting the pre-prepared lesson to the audience of students. Its is important that in such "tele-teaching " situation the hypermedia structure is not meant to be explored but to be presented. Second, a hypermedia document may contain self study material. This implies that the readers, i.e. the students, are confronted with a priory unknown information. This information is then not meant to be presented as in a tele-teaching situation but typically to be explored. In this context self study learning material adheres more closely to the real essence of hypermedia structures i.e. browsing of knowledge.

Bearing in mind the two user tasks, presentation and exploration, we are left with the question how learning material should be structured into networks of nodes. Kahn (1989) indicates how hypermedia systems eventually affects the meaning that is stored in hypermedia documents. The author reviews that in hypermedia systems like KMS and Hypercard, nodes are mapped as single screens or windows which are linked with uni-directional links from an object in one screen to another screen. The start of a link may have a fine granularity but the granularity of the destination of links is coarse. Systems like Guide and Intermedia provide a much finer granularity for the destination of a link, nodes are not restricted to a fixed screen but can consist out of multiple windows with flexible size.

In case of remote presentation of a lesson fine granularity of destination links is not strictly a necessity due to the fact that the teacher can use a telepointer to point to specific regions in a destination node. In self study learning material however, fine granularity of the destination of links might be preferable because of the lack of a teacher pointing to specific regions in a node.

Archive

In distance learning environments it is vital to store learning material in a central storage. Central storage, like databases installed on servers, provide the huge storage capacity needed to contain complete libraries of pre-prepared and/or un-prepared learning material. We distinguish here explicitly between pre-prepared and un-prepared or background learning material. Pre-prepared learning material are hypermedia documents which contain lessons for either tele-teaching or self study. Background learning material is the bulk of information which teachers can refer to in order to place lessons in a broader context. This background material will be pre-dominantly non-hypermedia material. For instance, a lesson about famous composers could refer to parts of the complete work of Beethoven.

Learning material should be stored in an information system or archive that allows effective searching strategies. This archive can be implemented either as a file system, a traditional database, or a hypermedia network. Eventually the choice for a specific information system will determine which searching strategies are possible. File systems enable users to perform only a primitive form of information retrieval which basically consist of traversing file names stored in a file tree. Traditional databases allow analytical search strategies, like Boolean keyword oriented procedures or query languages. Information systems build as networks of nodes containing the learning material allow users to retrieve information by browsing. In (Marchionini & Shneiderman, 1988) the authors review some aspects of information retrieval in hypertext systems. They conclude that browsing of knowledge, which is inherent to hypermedia, gives good performance in case of exploring new-task domains, but performance is rather low in case of fact retrieval. The authors point out that determining criteria for optimal mixes of browse and analytical support is critical to development.

We assume most students will enter a new-task domain when consulting learning material and benefit if the environment supports analytical searching strategies. However, experienced students and teachers will mostly explore familiar knowledge and will be best served by the hypermedia browsing capacities of an information system.

Preparation

The inertia of learning material may be identified as one of the drawbacks in traditional distance learning. New generation distance learning environments should not only provide the means for creating flexible or customizable learning material but above all they should lower the threshold for authoring. Semi structured nodes (Conklin, 1987), which are nodes that already contain pre-prepared templates may be particularly helpful towards generic or customizable learning material.

Specifically for preparing lessons, there is a need for multimedia object libraries. They are typically pre-prepared hypermedia documents containing various assortments of objects. For instance, an object collection may contain a large set of, hypermedia organized, sound samples of ancient instruments or contain a large set of digitized photos of various tropical birds. Object libraries differ from above mentioned background information in that they are not meant to refer to, but merely they are "cut and paste" material for hypermedia document preparation.

Annotation

It is vital in learning processes that students can comment their learning material. Therefore a hypermedia system should allow students to create new nodes and links in any instance. This would imply a modeless hypermedia editor which supports simultaneously browsing and the possibility to add new nodes and links.

Sharing

Inherent to groupware (Ellis, Gibbs, & Rein, 1991) are a "shared environment" and a "common task". In distance learning we may distinguish different common tasks and different shared environments. Basically all learning resources can be shared, that means both social learning resources, like teachers, and learning material can be shared among participants. Sharing learning material can take place on two levels. Sharing the central archive and sharing of hypermedia documents. Sharing the archive is less a groupware issue but more a time-sharing problem. Associated with it is the design of mechanisms for user identification and the possibility to assign read and write access to documents.

Co-authoring of learning material is a real groupware issue in that sense that it combines both the notion of a shared environment and a common task. The shared environment is a hypermedia document, or better, the public window of a hypermedia editor which instances appear on several workstations. The common task is preparation of a hypermedia document. In groupware taxonomy, co-authoring of hypermedia documents can be classified under multi-user editors.

Using OLE technology for Hypermedia: Some observations

Using OLE technology for hypermedia environment implies that we are somehow steered in our design decisions. We describe some experiences we had with OLE technology and place it in the context of the issues we mentioned previously.

Document

The impact that OLE technology has on the structure of an hypermedia document is visible in both nodes and links. An essential aspect of OLE technology is that we (the OLE client) can control the representation of an OLE objects but not the behavior, which is controlled and defined by an OLE server. This will affect the granularity of the destinations of hypermedia links. Our hypermedia editor is capable of making links with fine-granular destinations in so far it is within the graphical representation of an OLE object. In case of dynamic objects, like sounds or animation's, the destinations of links are limited only to the graphical representations of these objects. For example, we can point to the graphical representation of a sound or animated object but not to a specific location within such dynamic object.

Using OLE object excludes the "hypermedia node to one window mapping" approach in case we open (or play) an OLE object because it's associated OLE server will pop up in a separate window. Figure 4, shows an example of a typical HyDE transparency. Figure 4, at the left shows a background transparency with a combination of OLE text, OLE drawing, OLE image in combination with an superimposed overlaying transparency containing a "pie chart". The underlined words in the text, the button "Previous Page" and the question marks in the image are representations of anchors connected to background transparencies. Whereas the buttons, "Show Bars Graph" and "Show Pie Chart" are representations of anchors connected to overlaying transparencies. Figure 4, at the right shows how the OLE object, after a mouse click, is opened by its associated OLE server in a separate window.

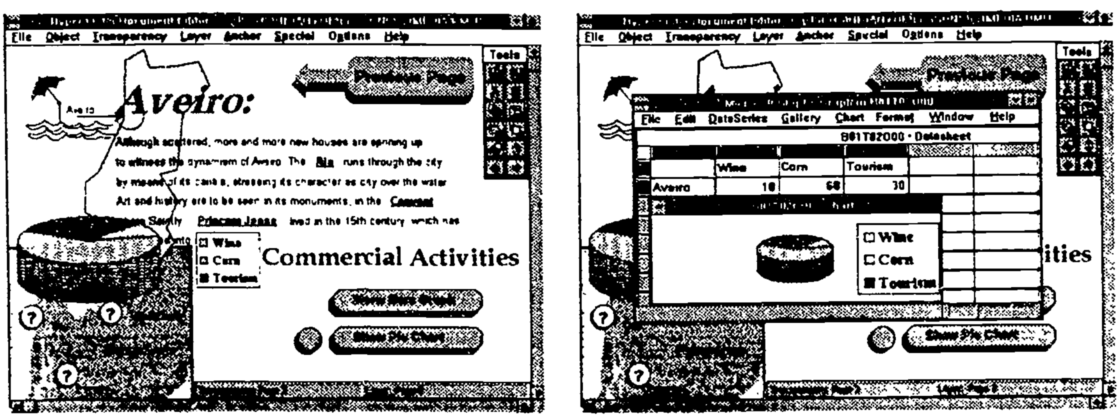


Figure 4. HyDE transparencies

Archive

HyDE documents can be stored in several databases. The structures of OLE objects are transparent for the database. Therefore, in order to allow effective search strategies, it is preferable to attach additional information to each OLE file. In HyDE, we provide the possibility to attach lists of keywords to HyDE lessons as well to individual OLE files before storing them on the server. These keywords typically describe the contents of lessons and OLE files and allow students and teachers to search for information with Boolean and keyword based procedures.

Preparation

OLE technology seems particularly suitable for constructing templates of multimedia objects. For example, a teacher can incorporate some blank spreadsheet cells made by a specific OLE server. Students can subsequently trigger the spreadsheet application by a mouse click and fill the blank cells.

With OLE, objects can be easily transferred from one document to another by the "cut, copy and paste" facilities of the MS Windows clipboard. This feature specifically will encourage the creation and use of multimedia object libraries.

Annotation

The ability for students to make annotations is not specifically influenced by using OLE technology. It is imbedded in the capabilities of the hypermedia editor to create new nodes and links while a reader is browsing the document.

Sharing

Currently all OLE servers are (to the best of our knowledge) single user applications. This implies that co-authoring in OLE based hypermedia editors is restricted to the level of composing hypermedia pages in the form of cooperative re-arranging of the graphical presentations of OLE objects.

Due to the fact that the format OLE objects are transparent for the archive, write and read administration can only take place on object level. This restrict us in the abilities, to differentiate between parts of an OLE object with or without write protection.

Discussion

We indicated several issues related to hypermedia learning material in distance learning environments and described the impact that OLE technology might have on each of these issues.

Both the advantages and limitations of OLE technology seem to be imbedded in its format or object independent approach. With OLE we can build open hypermedia environments which can benefit of, and anticipate on, a still growing spectrum of OLE editors. The price we pay for this object independence is that the structure and behavior of objects are transparent for the hypermedia environment. This will result in a lost in flexibility to create concurrent and synchronized behavior of OLE objects and a lost in flexibility in creating fine-granulated destinations of hypermedia links in case of dynamic objects. However, related to hypermedia learning material OLE technology appears useful because it allows fast preparation of courseware that is flexible for modification.

References

Borst Pauwels H.W.J., Pinto J.S., Sousa Santos B., Martins J.A. (1993). An Integrated Courseware Editor Based on OLE Technology. In , *Human Computer Interaction, Vienna Conference, VCHCI '93, Grechenig T., Tschelig M., (Eds.), Lectures Notes In Computer Science,* Vol. 733, Springer-Verlag, Berlin Heidelberg, 439-440.

Conklin J. (1987). Hypertext: A survey and introduction. *IEEE Computer,* 20, 9, 17-41.

Ellis C.A., Gibbs S.J., Rein G.L. (1991). Groupware some issues and experiences. *Communication of the ACM,* 34, 1, 39-58.

Kahn P. (1989). Webs, Trees, and Stacks: How Hypermedia System Design Effect Hypermedia Content. In, *Designing and Using Human-Computer Interfaces and Knowledge Based Systems,* Salvandy G. and Smith M.J. (Eds.),Elsevier Science Publishers B.V., Amsterdam, 443-449.

Marchionini G. and Shneiderman B. (1988). Finding facts vs. browsing knowledge in hypertext systems, *IEEE Computer,* 21, 1, 70-80.

Nielsen J. (1990). *Hypertext and Hypermedia*, Academic Press.

Pinto J.S., Borst Pauwels H.W.J., Sousa Santos B., Martins J.A. (1994). HyDE: a hypermedia document editor based on OLE technology, Accepted for presentation at: *1994 International Conference on Multimedia Computing and Systems*, Boston, USA, May 14-19, 1994.

(*) Work done as part of CO-LEARN (D2005), a project financed by the EEC within the DELTA program.

An Integrated Environment for Distance Education Supporting Multiple Interaction Styles

S. POZZI
CEFRIEL
Via Emanueli 15, I-20126 Milano, Italy - Tel. +39-2-66100083
Email silvano@mailer.cefriel.it

A. TANCREDI
Via Tribunali 40, I-87100 Cosenza, Italy - Tel. +39-984-21152

F. TISATO
Dipartimento di Scienze dell'Informazione
Via Comelico 39, I-20135 Milano, Italy - Tel. +39-2-55006253
Email tisato@hermes.mc.dsi.unimi.it

Abstract: The architecture of an open software platform supporting the activities of the educational process is presented. The architecture, which is based on a unifying data model, offers a set of services allowing to control co-operation activities based on either synchronous or asynchronous interaction styles. The software platform built on top of the proposed architecture can be used for integrating both specialized tools supporting specific activities of the educational process and general purpose tools. The paper introduces the general data model, discusses the cost-effective use of communication media in contrast with different co-operation styles, and exemplifies the proposed approach by illustrating two case-studies: an electronic mail-based system for distance tutoring *(TEMPO)*, and a software environment for real-time multimedia conferencing *(ImagineDesk)*.

Keywords: distance learning, synchronous co-operation, asynchronous co-operation, educational process, interaction styles

1. Introduction

The educational process consists basically of communications among the involved actors. First, it relies on the transfer of information from teachers to students (e.g. via "ex cathedra" lessons) and from authors to students (e.g. via textbooks). It also requires bi-directional communication among teachers and students to get feedback from the students, to support their assessment and to provide psychological as well as motivational support. The authoring activity implies communication among authors. Moreover, there are communications among actors involved in different activities of the overall process. Considering a waterfall model of the relationships holding among the phases of a typical educational process (from the definition of goals of the educational process, to the design of a specific course, to the learning phase of the course itself) we notice that people involved in the "higher" phases of the process (e.g., the definition of goals) must communicate with people involved in the "lower" ones (e.g., the design of a specific course). On the other side, feedback must be provided from the lower phases (e.g., learning) to the higher ones (e.g., from curricula definition to authoring).

The achievements of Information Technology (IT) may provide a valuable support for many of the activities and communication styles involved in the educational process. However, the development of IT-based educational systems requires the integration of expertise from two different domains, that is, *Education Science* and *Information Technology*, which are characterised by non-homogeneous theoretical and technological backgrounds. This may lead to systems exhibiting technologically advanced platforms, but poor educational models; in other cases, we observe systems featuring sophisticated educational models without an adequate technological support. Moreover, IT based tools associated to different activities constituting the educational process rely often on different models of the needed information, support different co-operation styles and are based on different media and technologies. As a result, information produced by a given activity can be hardly used as input for other activities. Ultimately, it is difficult to effectively manage the educational process as a whole.

This paper is focused on the design of an *integrated* environment supporting different activities and co-operation styles related to a given educational process. The environment defines a uniform model of the data, and uses in a cost-effective way multiple communication media. It is *open-ended* as it allows to smoothly integrate new tools and new media.

In Section 2 we sketch a model of both the educational process and the related data. Then we discuss the relationships holding between co-operation styles and communication media. The use of these concepts is exemplified in Section 3 through the presentation of two real cases. *TEMPO* is a distance tutoring environment which augments a mailing system by embedding it into a context supported by a Data Base of the learning activities. *ImagineDesk* is an environment for real-time conferencing integrating multiple communication media under a user-defined floor control discipline, and providing a platform for the development of multi-user educational applications. *ImagineDesk* provides also a support for planning the execution of co-operative activities. Finally, Section 4 shows how the achievements of *TEMPO* and *ImagineDesk* are going to converge towards a common platform aiming at supporting the overall distance learning process.

2. An Integrated View of the Educational Process

2.1. A Model of the Process

In our view, the educational process can be divided into six main phases: *requirements analysis*, *goals definition* [Mager 62], [Mager 76], [Stone 72], *design*, courseware *implementation*, *curricula management*, and *learning activity* . The goal of our work is not to define *the* ultimate model of the process; related concepts can be found, for instance, in [Carrol 63], [Skinner 68], [Bloom 76]. Our research efforts focuses on the definition of a unifying model [Tancredi 92] which is reasonable and simple enough to be the conceptual framework for the development of an integrated IT platform.

Though the phases are logically ordered, there is - or there should be - a complex and continuous feedback among phases, and inside each phase. It is supported by the *assessment* activity, which allows to measure the degree of achievement of the expected goals [Lindquist 65], [Cronbach 70], [Bloom 72]. It is the basis for evaluating the quality of each phase of the process, and for triggering corrections whenever needed. The assessment activity is performed at different phases and at different abstraction levels ranging from self-assessment to the assessment of the overall process. Note that, as soon as the assessment deals with the activities performed by a groups of actors, the data must be aggregated to generate useful feedback to the higher phases of the process.

2.2. A Uniform Data Model

The activities of the different phases, and the related tools, are often based on different models of the (logically shared) information. This lack of integration may arise either from logical reasons; or it may arise from technological limits. In order to achieve integration at a reasonable cost, all the phases of the process should rely on a uniform model of the data. The Data Model [Barrese 92] [Calabrò 92] is based on objects connected by relations. An "object" is an information structure characterised by the actions an actor can perform on it.

The basic objects model educational *tasks*, which can be hierarchically composed of depending tasks. For instance, a curriculum consists of courses which in turn consist of units. A task is associated to the definition of a *goal*. A goal may be associated to an *assessment procedure*. Tasks at the lowest level *(units)* are composed by *items* (a piece of text, a video, a figure, and so on). The execution of tasks is partially ordered according to two basic relationships: *propaedeutic*, meaning that the contents of a task are a necessary condition for another task, and *taxonomy* [Bloom 56] [Gagné 70] ordering the tasks according to their difficulty. An auxiliary *precedence* relationship may be defined to enforce a specialized path through the tasks.

The network of objects and relations of the data model defines the basic structure of the concrete Data Bases supporting specialized tools, and provides the framework for their integration. Note that this does not mean that all the activities and tools rely on a unique Data Base. What is needed is that all the Data Bases are designed according to the unifying data model, to facilitate the exchange of information among different phases and tools.

2.3. Communication Media and Co-operation Styles

Several different co-operation styles are involved in an educational process, and most of them may be supported by suitable IT tools. The problem is to exploit in a cost-effective way the available communication media in order to support the proper interaction style for each activity. As an example, in the following we provide a classification of the basic co-operation styles which can be identified in a specific educational activity, that is the learning one, and discuss briefly which communication media are the most suitable for each style.

Communication media fall into two major categories. *Real Time* communication is very effective, but is also very costly in terms of both technological and human resources (because it implies the availability of the actors at given times). *Non Real Time* communication is based on data repositories, supporting asynchronous co-operation [Ellis 91]. Its major advantage is the low cost; it is, however, less impressive and requires some degree of self-motivation.

A second distinction can be done between *symmetric* and *non symmetric* media. The former ones allow all the involved users to play an active role, whereas the latter ones reserve this right to some privileged actors. A major advantage of the symmetric media, which are more expensive, is the psychological and motivational support they provide. Broadcasting media (e.g. video lessons) are examples of real-time, non symmetric communication; conferencing systems are real-time and symmetric; mailing systems and repository systems are non real-time and symmetric; finally, textbooks are examples of non real-time, non-symmetric communication.

Co-operation styles can be classified in a two-dimensional space, whose dimensions represent the degree of prescriptive guidance and the degree of psychological support, respectively. The presented model is derived from models widely used in the area of situational leadership [Hersey 82] [William 67]. During the execution of an educational task, the preferred style for co-operation among students and teachers changes according to a typical path through the styles space.

At the beginning of the task there is a bare transfer of information and of directives from teacher to students. We call this style of co-operation *prescription;* the typical habit of the teacher in this style can be summarised as "that is what you must do (or know)". This style requires a limited degree of interaction among users, therefore it can rely on non-symmetric media: ex-cathedra lessons, distance lessons, textbooks.

As soon as the student encounters some difficulties in its task, he or she needs both prescriptive guidance and psychological support. The proper style in this phase is *explanation,* where the teacher's habit is "you want to do, and I'll show you how to do". The explanation style requires both fine grained guidance and psychological support, therefore real-time and symmetric media are needed. We argue that Real Time interactions require the integration of multiple real-time, symmetric media (text, voice, video, images...) under a common floor control discipline [CSCW 92, CSCW 93]. Most environments for real-time co-operation are built on the top of specific communication media, and do not provide the capability of integrating different media. For instance, it is not straightforward to integrate video conferencing with standard text editing tools in a co-operative environment. In general, the integration in the small should be achieved by means of a platform (see Sec. 3.2) that allows to design and implement co-operative environments by separating the logical aspects (e.g., floor control strategies) from the technological ones (e.g., the management of specific media).

In the next phase the student has achieved a reasonable degree of autonomy from the point of view of the contents of the task, but still needs some psychological support. The proper style is *involvement:* "I believe in your capability, and I am available to support you if you need help". The involvement style relies on a reasonable autonomy of the student, but requires psychological support. Therefore it can be based on non real-time symmetric communication, provided that the actors (in particular, the teacher) have a detailed view of the interaction context (this is the case of the tutoring activity which will be discussed in Section 3.1).

The last co-operation style is *delegation,* that should take place when the student is highly autonomous from both points of views: "the achievement of the goal is up to you". This style implies a high level of motivation and autonomy of the student, who may communicate with the teacher - if needed - via standard symmetric media: mail or telephone.

3. Practical Environments

3.1. TEMPO: an Environment for Distance Tutoring

TEMPO (TEletutoring, Milan, POlytechnic) is an environment supporting distance tutoring. It has been conceived and designed in order to support interactions related to the involvement interaction style, which can be based on non real-time symmetric communication, provided that the actors (in particular, the tutor) have a detailed view of the interaction context. In fact, the major difficulties concerning the communication among tutors and students arise from the lack of context. In particular, the tutor needs a view of the curriculum of the student, of the tasks he or she performed, and of the results of its assessment activities.

TEMPO is constituted of two (conceptually and physically) distinct work environments: the student and the tutor environments. They manage local Data Bases storing *images* of the basic objects and relations. The student DB includes the curriculum of the student, the history its interactions with the tutors, and the learning materials (or suitable references to them). The tutor DB includes the curricula of all the students, the learning materials and the histories of all the interactions. Both student and tutor environments rely on a MS-Windows platform. Student environments have a small local DB containing the student's data only. The tutor environments rely on a central DB supported by an Oracle DB server. The two environments communicate via e-mail, which is transparently used by *TEMPO* as a communication protocol to enable interactions among actors, and to keep the distributed DBs consistent.

TEMPO augments the e-mail system by embedding each help request into a *context* determined by the actual state of fruition of the learning material, so that the tutor can exactly know to which specific issue the help request is related to. Moreover, the tutor can analyse the curriculum of the student in order to customise the answer. Finally, on both the tutor and student sides a *history* of the interactions is maintained. It is supported by introducing the concept of *communication process,* which consists of a sequence of *communication acts.* The history of each communication process is stored inside the DBs as a time stamped sequence of typed communication acts. Users may navigate through a history, get the most recent acts of a process, get all the acts of a given type, select the acts related to a specific issue, and so on.

TEMPO has been presented here to show how a proper design of the data model provides a sound background for the integration of different activities during the educational process. The student environment supports learning and self-assessment activities. The tutor environment supports tools for tracing the interaction process, for collecting relevant questions, for the assessment of the overall process, and for managing it from an administrative point of view. Though the two environments are quite different, a high degree of integration can be achieved due to the fact that both of them are based on the same data model. The same principle should be applied to specialized environments associated to different phases of the educational process. For instance, information gathered from the tutoring activity (e.g., frequent questions) could easily provide feedback to the curricula management phase.

TEMPO has been developed at CEFRIEL under the sponsorship of the Polytechnic of Milan. A first version is currently used for the distance tutoring of a small (40) group of students in a basic course on Computer Science. An engineered version will be used during 1994 for larger groups.

3.2. ImagineDesk: an Environment for Real-Time Co-operation

ImagineDesk [DiNitto 93, DePaoli 91, DePaoli 93] is a software platform supporting the integration of different communication media in a real-time conferencing context. It can be used to support the explanation style of;co-operation . We define a conference as a system in which users interact in real time through a multimedia *shared workspace* [Ishii 91] constituted of a set of applications. For instance, a conference can be constituted by a spreadsheet, a word processor and a video-audio exchange application, so that interacting users can talk, see each other's face, and work on shared documents. *ImagineDesk* enables to organise and manage a conference by integrating both *co-operation-aware* applications (i.e., specifically developed to support co-operation) and *co-operation-ignorant* applications (i.e., standard off-the-shelf packages) under a common co-operation discipline.

Due to the fact that the collaboration takes place by interacting on a shared workspace, the access to the workspace must be carefully controlled by rules that must be customised to fulfil the requirements of specific activities. *ImagineDesk* considers that within a conference users are characterised by the *roles* they play during the evolution of the co-operative activity. A role defines the rights for performing actions on the shared workspace and for supervising the co-operation. Users playing the same role are collected in a *group*. Users may change their group membership in order to reflect changes of role.

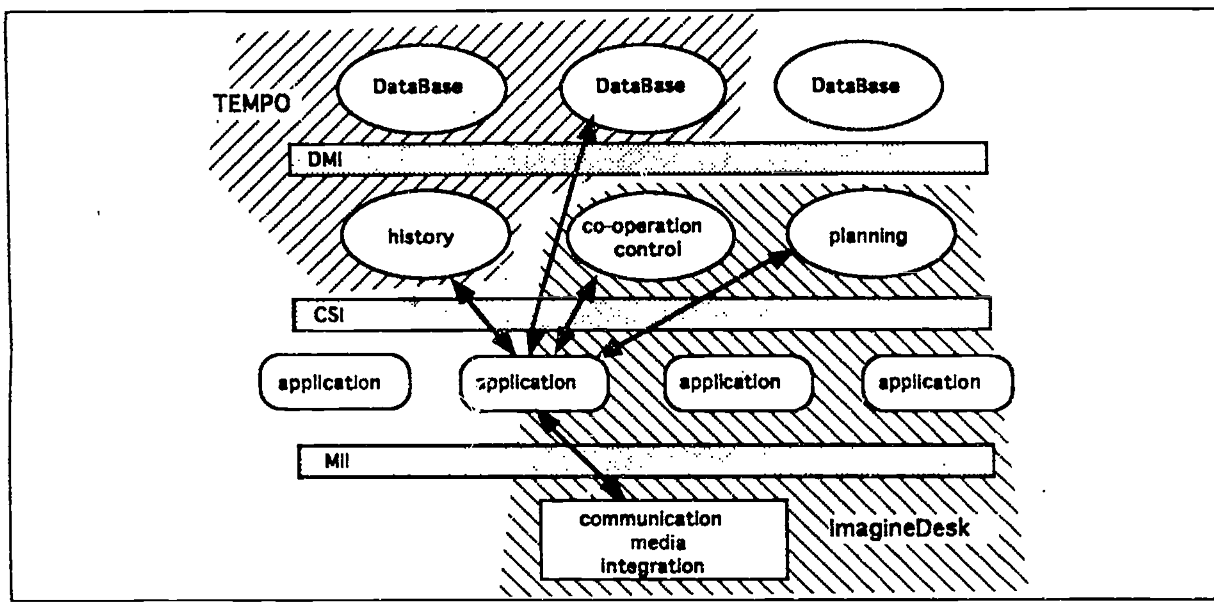
Groups are defined at two different abstraction levels. Groups at the *application layer* correspond to abstract roles that model the access to the shared workspace. Their definition is independent from user identity as well as from system-dependent features, such as communication media. Groups at the *communication layer* model the rights of access to the communication capabilities provided by the underlying hardware/software platform. Group membership changes at the application layer are reflected by corresponding changes at the communication one, according to a mapping relationships. Due to the clean separation between the logical and the physical aspect of the co-operative activity, it is possible to change its logical management without affecting the structure of the application, nor the communication aspects.

A key phase of a computer-supported co-operative activity (as well as of a real-life one) is the planning of the activity itself. In fact, at a great extent the accuracy of the planning phase (the scheduled start-up time, the set of invited persons, the context of work, and so on) determines the success of the co-operative activity. *ImagineDesk* provides for tools and services which can be exploited in order to effectively organise a co-operative activity. According to the interaction styles taxonomy presented in Section 2.3, these services can be used to properly organise the explanation of an educational topic, taking into account the users involved in the real time interaction, the media used to support the interaction, and the tools used through the interaction. Even during a real-time interaction users can refer to the actual context of fruition of the learning material, by relying upon the data model. For instance, as soon as an help process ends in a "cul de sac", that is, a student (or a group of students) is not able to exceed a specific problem, a real time interaction can be organised. The context of the interaction is the actual state of fruition of the learning material. Moreover, the tutor can decide to use one or more applications (such as word processor, drawing or simulation tools), in order to help the student solve the task, thus giving guidance and psychological support from a remote site.

The architecture of *ImagineDesk* includes two kinds of processes: *permanent* processes, existing as long as the platform is available, and *transient* processes, existing for a limited amount of time. *ImagineDesk* does not impose specific locations to the processes, unless a process controls specific hardware connected to a given machine. Processes communicate by means of asynchronous message passing. In a typical situation, co-operative applications run on User Workstations, while the processes comprising the platform run on the Conference Server, a workstation through which all control data flows and media flows are routed.

4. Conclusions: Towards an Integrated Environment

The achievements of *TEMPO* and *ImagineDesk* provide a sound basis for the development of an integrated, distributed and open-ended environment supporting the whole educational process. The final goal of the research activity is the definition of a software platform allowing the implementation of specialized components (courseware packages, assessment procedures, co-operative applications, authoring tools, administration services, and so on) on the basis of a uniform framework. The architecture of the environment is shown in the figure. The environment provides a collection of service modules supporting general purpose functionalities related to the communication among users: planning, history, and co-operation control. These modules are viewed by application programs through a suitable application programming interface (*Co-operation Services Interface*, CSI) hiding system dependent issues and providing a uniform view of the services.



The Integrated Environment.

The basis for the integration of specialised applications is provided by a common interface to Data Base services (*Data Model Interface*, DMI) based the general data model, no matter how and where it is implemented in terms of specific Data Bases. Both service modules and specialized tools rely on DMI.

Finally, the integration of different communication media is provided by a *Media Integration Interface* (MII) that provides a uniform visibility of the media.

As shown in the figure, all these issues are tackled by either *TEMPO* or *ImagineDesk*. The data model of *TEMPO* provides a general framework for the integration of tools devoted to specific activities and for the feedback among the phases of the education process. *TEMPO* supports also the concept of history of communication processes, that could involve different activities and tools. On the other hand, *ImagineDesk* provides a valuable support for planning and managing the activities, for integrating different applications under a common discipline, and for integrating different communication media in the context of an application.

Ultimately, the authors believe that the bricks are here, and that the integrated environment is partially built. Further efforts will be devoted to its experimental use, to its evolution towards an engineered platform, and to the development of specialized tools. The open-ended approach of the project allows to devise a stepwise strategy for the implementation of the overall environment. The experimental use of *TEMPO* will allow to test the consistency of the data model, to define the logical structure of a significant collection of courses, to collect and organise data related to a large group of students, and to experiment the practical use of the tutoring environment. This activity has been already started and will have a first milestone at mid 1994. Meanwhile the definition and experimentation of tools supporting assessment will be carried on, in order to support feedback in the learning activity. At this point, the skeleton of a technological, organisational and methodological framework will be available.

In parallel, the *ImagineDesk* platform will be enhanced by integrating new communication media (in particular, ISDN for supporting multimedia interaction over public networks), by extending the mechanisms for the definition of flexible co-operation strategies, and by building user-friendly interfaces for the co-operation control. Moreover, some experimental courseware packages for real-time, multi-user teaching/learning activities will be developed and tested.

Further steps will be oriented to the definition and development of specialized tools for curricula definition, authoring, and management of the overall process. Courseware packages will be gradually integrated as soon as they will become available; the value of the environment, however, derives mainly from its ability to provide an integrated framework from the organisational and methodological point of view, and is not heavily bound to the availability of complete sets of courseware packages.

References

[Barrese 92] Barrese R.M., Calabro' G., Gallo T., Tisato F., *CAMCE: An Environment to Support Multimedia Courseware Projects*, Educational & Training Technology International, 29,1,1992

[Bloom 56] Bloom B.S., *Taxonomy of Educational Objectives. cognitive domain*, McKay, NY, 1956

[Bloom 72] Bloom B.S.,Hastings J.T.,Madaus C.F., *Handbook on formative and summative evaluation of student learning*, McGraw-Hill, NY, 1972

[Bloom 76] Bloom B.S., *Human Charcteristics and School Learning*, McGraw-Hill, NY, 1976

[Calabrò 91] Calabrò G., Cozza S., Gallo T., Tisato F., *The CAMCE Authoring Environment*, Int. Workshop on Computer Aided Leaming and Simulation Technology, Prague, June 1991

[Carrol 63] Carrol J.B., *A Model of School Learning*, Teacher College Record, 1963.

[Cronbach 70] Cronbach L.Y., *Essential of Psychological Testing*, Harper-Row, NY, 1970.

[DePaoli 91] DePaoli F., Tisato F., *A Model for Real-Time Co-operation*, 2nd European Conf. on Computer Supported Co-operative Work, ECSCW91, Amsterdam, Sept. 1991

[DePaoli 93] DePaoli F., Tisato F., *Language Constructs for Cooperative Systems Design*, European Software Engineering Conference, Sept. 1993.

[DiNitto 93] Di Nitto E., Pozzi S., *ImagineDesk: a software platform supporting cooperative applications*, submitted to ACM-Conputer Science Conference (CSC) '94, Phoenix (AZ).

[Ellis 91] Ellis C.A., Gibbs S.J., Rein G.L., *Groupware: some issues and experiences*, Communications of the ACM, v. 34, n° 1 (1991).

[Gagné 70] Gagne' R.M., *The Condition of Learnig*, Holt Princhart & Winston, London, 1970

[Hersey 82] Hersey P.,Blarchard K., *Management of Organizational Behavior*, Prentice-Hall, Englewood Cliffs, 1982

[Ishii 91] Ishii H., Miyake M., *Toward an Open Shared Workspace; Computer and Video Fusion Approach of TeamWorkStation*, Communications of the ACM, Vol. 34, N. 12, pp. 36-50, December 1991.

[Lindquist 65] Lindquist E.F., *Educational Measurement*, Houghton Miffin, Boston, 1965

[Mager 62] R.F. Mager, *Preparating Instructional Objectives*, Fearon, Palo Alto, 1962

[Mager 76] R.F. Mager, *Goal Analysis*, Fearon Publishers/Lear Siegler, Belmont, 1976

[Object 91] Object Management Group, *The Common Object Request Broker: Architecture and Specification*, OMG Document 91.12.1, 1991

[Pozzi 93] Pozzi S., Giachino L., *An Object-Oriented Programming Environment for Multimedia Cooperative Applications*, submitted to Conference on Programming Languages and System Architectures, ETH Zurich (Swiss), March 1994.

[Skinner 68] Skinner B.F., *The Technology of Teaching*, Appleton Century Crofts, New York,, 1968

[Stone 72] Stone E., *Educational Objectives and Teaching of Educational Psychology*, Methuen, London, 1972

[Tancredi 92] Tancredi A., Tisato F., *Tecnologie Informatiche e Processo di Istruzio:ne un Approccio Integrato*, Internal Report 95/92, Università di Milano, DSI, 1992 (in Italian).

[William 67] William R.J., *The 3-D Management Style Theory*, Training and Development Journal, 4/1967

Acknowledgements - TEMPO has been developed with the sponsorship of the Polytechnic of Milan. The authors are indebted to Prof. Attilio Costa and to Prof. Carlo Ghezzi for their suggestions and their support.

Case-Exercises, Diagnosis, and Explanations in a Knowledge Based Tutoring System for Project Planning

MICHAEL PÜLZ, MARKUS LUSTI
Institut für Informatik
Universität Basel, Petersgraben 51, CH-4051 Basel, Switzerland
email puelz@urz.unibas.ch

Abstract: PROJECTTUTOR is an intelligent tutoring system for project planning. As a problem solving monitor, it completes conventional classroom teaching by training problem solving in project planning. The teacher uses the authoring component of the system to specify case problems. A case problem consists of a text describing the problem and a framework describing the structure of the problem. On the basis of an author-defined case text, the authoring component constructs a formal representation (an MPM-network), representing the structure of the problem. The student works with the tutoring component of PROJECTTUTOR. The text of a case problem is presented and the student has to find the structure of the project, build up an MPM-network, and perform time and cost analysis. The system compares the student's solution steps with those generated by the expert module. At any time, PROJECTTUTOR can give diagnosis, hints, and explanations, referring to the domain knowledge of the expert module as well as to the specific case.

Research objectives

The purpose of our research is to build a knowledge based tutoring system to improve *problem solving* in project planning (Pülz & Lusti, 1994). PROJECTTUTOR is able to diagnose student errors and to explain the solutions generated by the expert module. It explicitly uses information in the problem text to enhance explanations. In Operations Research, textbooks usually include case studies. The students solve the exercises using paper and pencil. Eventually they can check their own solutions against the solutions found in the appendix. Textbooks do not give any hints, explanations or diagnoses on the student's work.

PROJECTTUTOR can be used in any environment where project planning is taught. In our opinion, Operations Research is a well suited domain for tutoring systems. The curriculum is stable, structured, and algorithmically tractable (Angelides & Doukidis, 1990). In a study by Belling-Seib, the use of computer technology in the curriculum of Operations Research in Germany, Austria and Switzerland was investigated (Belling-Seib, 1991). It shows that Operations Research is a field where teachware is already used frequently, although most programs are just tools for computing solutions of given models. The purpose of PROJECTTUTOR is to increase the students' ability to develop a formal model from a given case problem and to work on the model until the problem is solved in the intended way.

The curriculum

Figue 1 describes the educational goals of PROJECTTUTOR. As prerequisites the students need the basics of graph theory and project planning. The domain knowledge covered by the expert module is devided into three parts. The *structural analysis* identifies the activities that make up the project. Each activity is associated with an average duration and optionally minimum duration and crashing costs. Next, the connections between the activities are identified and drawn using directed arcs connecting pairs of activity nodes. There are different types of connections (start-start, start-end, end-start, end-end), depending on whether a connection refers to the start or finish times of the associated activities. MPM-graphs usually use the start-start connection type. Connections can either be positive, in which case they are labelled with a positive number, saying that an activity

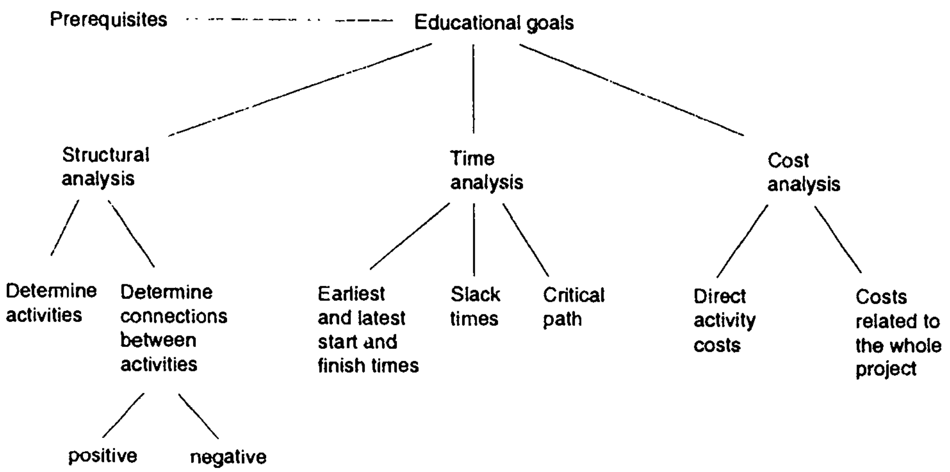


Figure 1: Hierarchy of educational goals

can start *no earlier than* a certain amount of time after the start or the end of the preceding activity, or they can be negative when labelled with a negative number. A negative number indicates that the *latest start* of an activity is a certain amount of time after the start or the end of the preceding activity.

Time analysis computes the earliest and latest start and finish times of each activity, their slack (float) times, and the critical path(s). *Cost analysis* computes the minimum project costs by using the slack times of the non-critical activities or determines the minimum project time and the project costs by crashing all the activities. Different types of costs, such as direct activity costs, costs associated with the whole project (both of which are variable costs), and costs that are not relevant for the crashing of activities (fix costs), have to be identified.

The system

PROJECTTUTOR consists of two subsystems, the *authoring* and the *tutoring component* (Lusti, 1992). The intention of the authoring component is to make the system more flexible. The teachers can specify their own case-exercises. The authoring component is the knowledge acquisition tool of the system. In PROJECTTUTOR only case knowledge can be acquired. The domain knowledge of the expert module is compiled and therefore static. The tutoring component is used by the students to work on the author defined case-exercise.

Knowledge acquisition and internal representation (authoring component)

The authoring component allows to edit the case base. A case-exercise is first described by an unstructured text. The teacher then overlays the text by marking relevant pieces of text with different colours. Similar ideas of analysing the information contained in prescriptive texts by using NLP methods have been described before (e. g. Moulin & Rousseau, 1990). Our system does not use NLP methods. This is because the author is free to type any text he/she wants. In particular, no keywords have to be used. Analysing such texts automatically is still very difficult. Instead, the author has to explicitly tell the system which piece of text contains which information. In the following example, the bold piece of text indicates an activity: "The **planning of the gardens** can start no earlier than 10 days after ..."

As an example, figure 2 shows a problem text in the window on the left side of the screen, including a marked text piece denoting a negative connection. Each marked piece of text has one of the following meanings: activity, average duration, minimal duration, crashing costs, connection, time requirements of a connection, variable project costs, or fixed project costs.

Marked text pieces are immediatly transformed to an internal representation reflecting the project structure (see figure 3). The representation is a list of text pieces that provides for the insertion of attributes like "activity", "duration", or "connection". Moving text preserves the attributes. Deleting text is only possible under *integrity constraints*. If, for example, the only piece of text about a certain activity is deleted, the system warns the author. He/She can then specify another piece of text or remove the activity from the network.

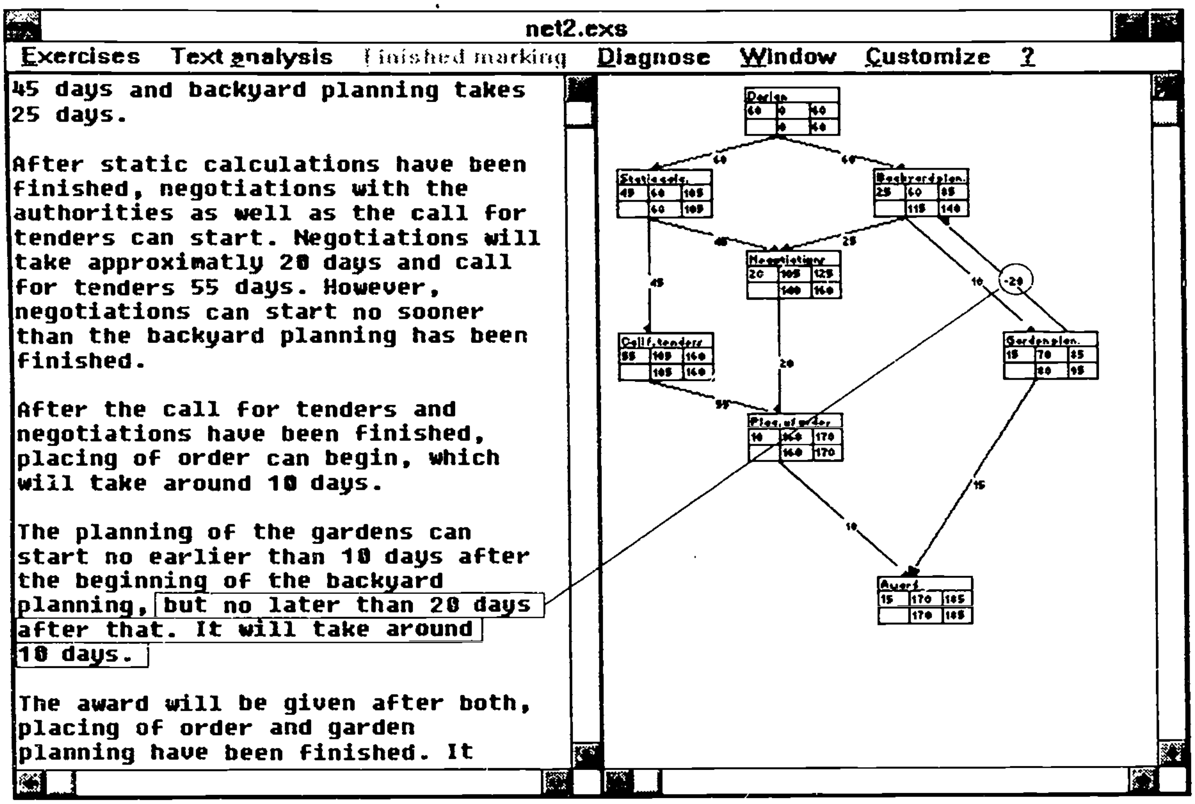


Figure 2: Knowledge acquisition using the authoring component

After adjusting the internal datastructure, the system automatically updates the *formal representation* in the network window on the right side of the screen by generating a corresponding activity-object (a node), a connection between a pair of objects (an arc), or time and cost data (see figure 2). The author can see the effects of his/her text analysis on the graph. Furthermore, the system detects inconsistencies within the author's definitions. The internal datastructure is stored in the case base and is used by the explanation facility of the tutoring component.

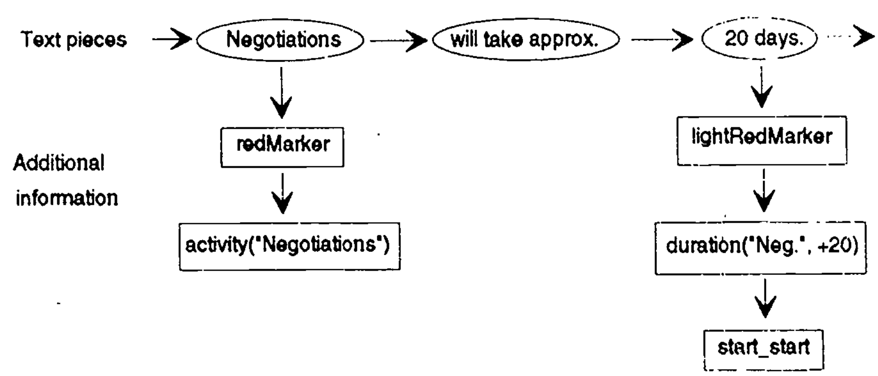


Figure 3: A part of the internal datastructure

Tutoring component

When the students work on case-exercises, PROJECTTUTOR does not explicitly lead them through a chain of tutoring steps, but they have to decide by themselves which step to take next. However, there is an implicit tutoring process since the students have to follow a certain sequence of tasks in order to arrive at the correct solution. At any time the students can ask the system to: diagnose the work the students have done so far, give hints for further solution steps, check whether a (sub-)solution is correct, explain why a (sub-)solution is not correct, or produce a (sub-)solution.

PROJECTTUTOR tries to behave like a human tutor who is available while students work on a problem. The students can ask the tutor and the tutor will give comments, hints, or explanations. The tutor has to decide which answer to give. In the current version, no explicit pedagogical decisions are made. Instead, every question that the students ask generates a local reaction of the system, consisting of a how-explanation trace (a tree) and references to relevant information in the problem text. By navigating through the tree an explanation dialog is carried out and links to the exercise text are shown.

Diagnosis means checking for discrepancies between the student's and the system's solution. This can be done at any time and consists of determining the current state of the student's work and checking the student's current (sub-)solution against the one generated by the expert module. If an error is detected, the system finds out whether the error occured because of incorrect network construction or incorrect transformation of the network.

Incorrect network construction can be due to misinterpretation or overlooking of information contained in the problem text. Students might forget to draw an activity node or a connection between two activities. Or they draw a connection in the wrong direction or between the wrong nodes. The system points out the misinterpreted or overlooked information in the problem text, allowing the student to improve the network.

Errors in transforming the network can be caused by different kinds of mistakes. There are different cognitive levels involved. An error might simply be due to a wrong arithmetic calculation, which is on a low cognitive level and can be easily recognised by the student. The system can simply give a hint pointing out the error.

At a higher level, project-planning algorithms to be performed on the network might be carried out incorrectly. Those algorithms are, for example, computing the earliest and latest start and finish times, the slack times, and the critical path within the time analysis. Here it is necessary to trace back the operations performed by the student. PROJECTTUTOR can then mention the erroneous step. The student might either recognize his/her mistake and repeat the relevant steps or, if there is a problem in understanding the algorithm, the student can ask the system to demonstrate the algorithm step by step. If necessary, the student can invoke the hypertext help facility to look up textbook knowledge explaining the algorithm and domain terms.

If the system detects that the student made the mistake because of overlooking or misinterpreting information in the exercise text, it refers to the corresponding text pieces.

The *explanation component* comprises explanations generated by the domain knowledge of the expert module and the knowledge about the meaning of the text stored in the internal representation of the problem text. As many references as possible are made to the problem text. If the student makes a mistake and asks for a diagnosis, the system responds by highlighting a relevant piece of text in the case text and displaying additional information.

In the following, a dialog between a student and the system is outlined. Computer output is written in italics and comments are put in brackets. (In fact, the dialogue is menu driven):

Why is the latest start time of the garden planning not 155 days?
(The student asks for a diagnosis and the system highlights the occurrence of a mistake in the network.)
Because there is a mishap in the structural analysis.
(The system realized that the mistake is caused by an error in an earlier part of the educational goals, which is the structural analysis; see figure 1.)
Why?
(The student asks for a more precise explanation.)
The connections in your network are not correct.
(The system moves to the next level in the explanation tree, but still keeps the explanation somewhat general. The explanation resembles a hint.)
Why?
(The student asks for further explanations.)
Look at the following information in the problem text: "... The planning of the gardens can start no earlier than 10 days after the beginning of the backyard planning, ***but no later than 20 days after that...****".*
(The system highlights the corresponding piece of information in the exercise text which was overlooked by the student. Here the explanation trace is explicitly connected to the problem text.)
How do I specify such a relation in the network?

(The student does not know how to incorporate this information in the network and asks the system for further help.)
You should use a negative connection.
(The system reached the lowest level in the explanation tree.)
What is a negative connection?
(The student does not know that term and asks for more help.)
A negative connection is...
(The system starts the hypertext help facility.)

We believe that *explanation traces* generated by the domain knowledge of the expert module should be *accompanied by references to the problem text*. Other research comes to similar conclusions (Reiser, Ranney, Lovett & Kimberg,, 1989). Studies on mathematical text problem solving found that unstated or hidden relations in the problem texts are a major source of errors (Nathan, 1993). A system should also be able to present different representations of the problem. PROJECTTUTOR provides two views of a case problem. One is the unstructured exercise text and the other is the formal network representation.

Communication module

The communication module of PROJECTTUTOR is crucial for the acceptance of both its authoring and tutoring component. The nodes and arcs can be moved around by dragging and dropping. At any time, the students can ask questions. *Direct manipulation* lets the students (as well as the teacher) see the effects of their work immediatly (LeBlanc, 1993). If the students specify a link between two activities, this link is drawn immediatly in the network. The same is true after an author has marked a piece of text indicating a link between activities.

Matching the student's work with the system's work

As described earlier, the curriculum is devided into different parts (see figure 1). There is a certain order in these curriculum parts. For example, the student has to correctly finish the structural analysis before carrying on with the time analysis. However, the system allows the student to work on the time analysis, even if the structural analysis has not been finished correctly. The idea here is to let the student find out about the error in the earlier phase (*explanation by contradiction*; Rätz, 1993; Rätz & Lusti, 1992). The student then has to go back to the structural analysis and correct it. Since the time analysis is based on the structural analysis, the work that has already been done in the time analysis is lost. The student has to redo work that he/she might have already proved to know. In those cases, PROJECTTUTOR can do this work for the student. This is in order to keep the student concentrated on crucial problems and to automize minor ones.

When working on project planning exercises using paper and pencil, the students often have to recompute a whole network after they realized that the structure of the network is not correct. This is tiresome and demotivating.

PROJECTTUTOR keeps track of the student's work in a simple way. Once a subtask has been completed correctly, e.g. computing the slack times, it is marked as known by the student. If the same subtask has to be performed again at a later time, PROJECTTUTOR offers to do it for the student.

PROJECTTUTOR also compares the student's work with the one done by the expert module (see figure 4). A matching algorithm is implemented that detects missing, wrong, and incorrect items in the student's work. Based on this information, diagnosis and explanations are built up.

Discussion and further refinements

We described PROJECTTUTOR, a monitor for training problem solving in project planning. The curriculum and the levels of competence were mentioned. We discussed the different moduls of the system, stressing the knowledge acquisition part, the tutoring component, and the communication module.

PROJECTTUTOR includes an authoring component which offers two consistent representations to the teacher: the natural language text and the graphical network representation. The tutoring component provides a powerful explanation facility (incorporating the problem text and the explanation traces generated by the expert module).

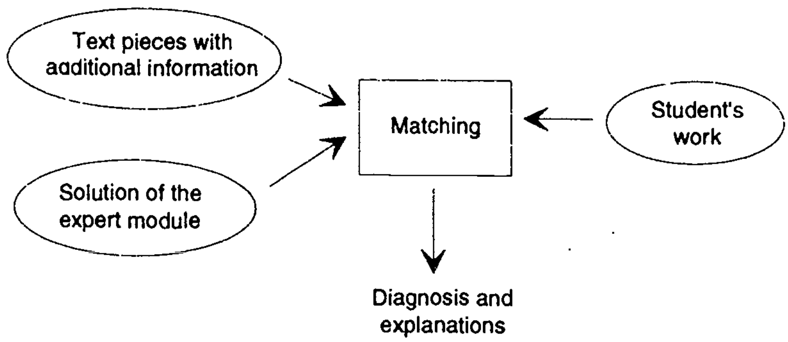


Figure 4: Matching the student's and the system's work

A qualitative evaluation was carried out, showing that the students appreciated the advantages of the system compared to pure paper-and-pencil exercises. However, further improvements have been pointed out in several directions. An extension of the curriculum to include CPM- and PERT-techniques, capacity-, milestone-, and trend-analysis would improve the usefulness. Case problems could be extended to vary during the exercise, e. g. the system could tell the student that a certain activity is delayed and the student has to react. The system would then become more of a simulation tool. Student modelling could be extended and pedagogical knowledge could be added so the system can adapt better to the individual student.

PROJECTTUTOR is written in PDC Prolog and runs under Windows 3.1.

References

Angelides, M. C., Doukidis, G. I., Is There a Place in OR for Intelligent Tutoring Systems?, *Journal of the Operational Research Society* 41 (6) 491-503, 1990

Belling-Seib, K., Educational OR software/Computer assisted OR education, *Methods of Operations Research* 64, 645-647, 1991

LeBlanc, M. D., From natural language to mathematical representations: A model of 'Mathematical Reading', in NWAN93, 126-144

Lusti, M., *Intelligente Tutorielle Systeme: Einführung in wissensbasierte Lernsysteme*, Bd. 15.4 des Handbuchs der Informatik, München 1992, Oldenbourg

Moulin, B., Rousseau, D., Structuration d'une base de connaissances a partir du contenu de textes prescriptifs, in *Tenth International Workshop Expert Systems & their Applications, Specialized Conference on Natural Language Processing & ITS Applications*, Avignon, May 28-June 1st, 1990

Nathan, M. J., A simple learning environment improves Mathematical reasoning, in NWAN93, 162-186

Nwana, H. S., *Mathematical intelligent learning environments*, Oxford 1993, intellect

Pülz, M., Lusti, M., *Ein wissensbasiertes Lernsystem zur Projektplanung*, WWZ-Discussion Paper, Nr. 9402, Universität Basel, Basel 1994

Rätz, T., *Erklärungen in wissensbasierten Lernsystemen am Beispiel eines Tutors zur Normalisierung von Datenbanken*, Frankfurt 1993, Lang

Rätz, T., Lusti, M., Explanation Strategies: Realization in a Tutor for Database Normalization, in Brézillon, P. (ed.), *Proceedings of the European Conference on AI, Workshop W15: Improving the Use of Knowledge-Based Systems with Explanations*, Institut Blaise Pascal, Université de Paris, Rapport 92/91, 1992 47-56

Reiser, B. J., Ranney, M., Lovett, M. C., Kimberg, D. Y., Facilitating Students' Reasoning with Causal Explanations and Visual Representations, in BIER89, 228-235

Computers as Interactive Representational devices for Declarative domains

HELEN PURCHASE

Computer Science Department, University of Queensland, Australia

Abstract: This paper reviews semiological theories in relation to the use of computers for enhancing learning activities, concentrating particularly on the use of graphical (or *schematic*) texts in declarative domains. Relevant semiological and educational issues are discussed, and three areas where the use of schematic representations in computer learning environments may be investigated are proposed: representing a complex structural domain, depicting the lexical relationships in a thesaurus, and assisting the authoring of an interactive fiction system. Three existing systems which address the associated learning activities are also described.

Introduction

Literacy is a primary aim of education: subject matter is usually taught 'out of context' with the use of various forms of text, and it is considered essential that children learn the skills to decode represented experience competently so that the information and techniques which form part of a culture may be propagated. Emphasis is therefore placed on reading, numeracy and the interpretation of diagrams and film. The computer is a significant addition to this list: its power as a representational device is greater than other media because of its interactive and processing nature. This paper considers the use of computers in education from the point of view of representational semiotic devices. First, relevant semiotic terminology is defined and appropriate educational issues addressed. Three projects which entail the design and development of computer educational systems which use an interactive representational method are then described. The aim of each of these systems is to investigate the suitability of using schematic representational methods for the depiction of declarative subject matter in an interactive device. They are each related to a particular learning activity: envisaging the structure of a complex domain, discovering the relationships between words in a lexicon, and authoring a hypertext system.

Semiology: terminology and classifications

Definitions

Semiology was defined by Saussure in 1915 "*A science that studies the life of signs within society*...[to] show what constitutes signs, what laws govern them..." (Saussure, 1959). While his emphasis was primarily linguistic, Peirce's analysis of signs (which he termed *semiotics*) was more general and concentrated on the characteristics of all signs (Peirce, 1977). Terminology differs between Saussure's and Peirce's definitions of signs and symbols, and for the purposes of this document, Peirce's definitions and categorisations will be used; that is, a *sign* is an intimate relation between an object (*term*) and an interpretant (*concept*). While the nature of the bond between the two components of a sign is irrelevant, a *symbol* is a particular category of sign, where the relationship between its object and interpretant is arbitrary. A *semiotic system* (*code*) is an organisation of patterns of particular signs (usually rule- and convention-based) that comprises a system of meaning, and a *symbolic system* is a type of semiotic system. A semiotic system thus consists of a *syntax* which defines the manner in which terms may be organised, and a *semantics* which indicates how meaning can be attributed to a syntactically correct pattern of terms. A *statement* is defined as a syntactically correct and meaningful combination of terms in a semiotic system, a *message* as a semantically coherent sequence of statements, and *text* (*representation*) as the physical realisation of a statement or message. A *representational system* (or *medium*) is an abstract term that refers to the physical realisation of the rules and conventions that comprise a semiotic system, and a *device* is a physical object used for communication via a semiotic system. *Communication* takes place when text is created according to a particular code, and transmitted via a device. The receiver of the text decodes it to extract meaning, choosing to impose a particular code on the text to enable interpretation.

For example, music is a *semiotic system* comprising *signs* (notes), where each note consists of a *term* (*eg* E flat) and a *concept* (the sensation produced by the corresponding sound wave). A figure from Beethoven's ninth symphony is a *statement*, the whole symphony is a *message*, the score is a *text*, an orchestra is a *device* for communicating it, and musical notation is the abstract term denoting the corresponding *representational system*.

A distinction may be made between *non-interactive* communication devices, which do not allow the receiver of a text to interact with then (*eg* televisions), and *interactive* communication devices, whose flexibility allows text to be altered and manipulated by the receiver. The most obvious example of an interactive communication device is a computer: its flexibility allows for dynamic transmission and manipulation of signs and texts.

Classifications

Bruner identifies three different types of semiotic systems: *enactive*, *iconic* and *symbolic*, with the corresponding signs *action*, *image* and *symbol* (Bruner, 1966). The enactive system is based on physical movement and the learning of responses (*eg* riding a bicycle), the iconic system depends on imagery and perception (*eg* looking at a picture), and the symbolic system uses symbols which do not have a perceptual relationship with the objects they signify (*eg* reading a sentence in natural language). For the purposes of this document, symbolic systems are further classified into *schematic* and *linguistic* categories, the main difference between them being the way in which the symbols are physically organised. Schematic systems use spatial indicators to show the structure of the subject matter (*eg* circuit diagrams, flowcharts), while linguistic systems place the symbols in a purely sequential manner (*eg* formal logic, morse code). In both systems, the way in which the symbols in a text are organised is important: linguistic systems have a linear syntax which determines how the meaning of the separate symbols unite to produce the semantics for the whole statement. Schematic systems tend to be graphical in nature, the overall structure of the subject represented and the relationships between the individual symbols being interpreted according to a conventional code (*eg* straight lines in circuit diagrams represent physical connections which are not necessarily straight). The physical realisation of a statement or message in a schematic system is here termed a *diagram*. Schematic representations are exceptionally useful. Some concepts cannot be captured by icons or individual symbols alone and the holistic nature of diagrams ensures structure and relationships are made explicit, often making a simple diagram more effective than the equivalent linguistic form.

Semiotic skills and the educational use of computers

The use of semiotic systems is mankind's most distinctive characteristic; coded experience and effective semiotic systems have proved essential for the propagation of knowledge. In education, communication via semiotic systems forms an important role in guiding children towards the effective use of information and techniques, and it is therefore important that children be able to decode a given representation competently; they should know the appropriate code well enough to both understand the text as well as manipulate and use it. For this reason, the development of literacy in various semiotic codes is the primary concern of much formal schooling.

Subject matter is usually taught to children 'out of context' by the use of a (usually symbolic) medium. Olson and Bruner make a clear distinction between the knowledge represented in a text and the skills required to decode the text. They consider that, regardless of whether the medium is enactive, iconic or symbolic, a common invariant 'deep structure' of knowledge can be created. However, the skills used for extracting and utilising this knowledge in each medium differ greatly, and they recommend that this should be considered when a representational system for instruction is chosen:

> The choice of a means of instruction, then, must not depend solely upon the effectiveness of the means for conveying and developing knowledge; it must depend as well upon its effects on the mental skills that are developed in the course of acquiring that knowledge (Olson and Bruner, 1987).

The choice of medium for encoding subject matter for educational purposes is therefore an important issue on two counts. Firstly, the ability of the intended users to understand the code in order to perform the required task should be considered. This is related to the cognitive development of the children and their previous experience with similar systems. Secondly, the suitability of the medium for the subject matter embodied and the intended learning activity ought to be addressed: inappropriate use of a medium may result in the creation and use of unsuitable and confusing mental models of the domain.

Both of these issues are made more complex when the computer is chosen as the medium of instruction. As semiotic devices, computers have additional power in that not only can they externally *present* representations, they can also internally *manipulate* them. Educational studies regarding the semiotic nature of the computer tend to be concerned with the cognitive benefits accrued from programming (*eg* understanding random numbers, recursion, variables *etc*) (McDougall, 1990). They have concentrated on the symbolism afforded by programming languages (Papert, 1980; Pea, 1987), and have not addressed the computer's potential as a semiotic interactive device for the explicit representation and manipulation of information, where no enactive (*ie* simulation or program) counterpart exists. The emphasis in the practical educational use of computers is usually on the subject matter embodied within the software, or on the computer skills re-inforced by computer use, rather than on the nature of the medium used.

Three interactive schematic systems

In focusing on the semiotic nature of the computer as a representational device, educational researchers and practitioners ought to consider whether its interactive nature contributes in any way to the learning activities that the user is engaged in (or indeed, whether it inhibits them). The research reported here considers interactive declarative schematic representations in particular, and three learning activities have been identified as potential areas where their use may enhance learning: the representation of complex structural domains, the navigation of a 'lexical space' depicting the relationships between words, and the display of a hypertext structure in an authoring tool.

Three systems have been designed to investigate the use of interactive schematic representations for these learning activities: they are described briefly below, with particular emphasis given to the nature and use of each the schematic representations used.

Schematic hierarchical knowledge base structures

KREEK is an educational learning environment which embodies an interactive schematic representational method for the portrayal of structured declarative domains (*eg* the solar system, an animal taxonomy), using a network syntax where *nodes* represent objects and *links* represent relationships. Relationships between the main objects in a text are made explicit by forming links between them, while attribute (or 'property') information associated with the objects are encapsulated within *frames* at the nodes (Minsky, 1975). In addition, each node may have associated with it a *layer*, which is itself a syntactically valid diagram and covers the entire computer display. Thus KREEK texts may consist of any number of layers organised in a hierarchical structure, allowing the knowledge-base to be structured so that subordinate layers provide more specific information about an object.[1] The dynamic environment offered by the computer is therefore used in order to incorporate structural conventions: not all of the information is made available at once, and the parts of the text that are not initially visible (either in frames or in layers) may be selected by the user for further perusal. KREEK may be used for both browsing and creating these hierarchical schematic texts: the intention is that the inherent structure of a knowledge base may be better envisaged by making it explicit in an interactive schematic text.

Qualitative preliminary evaluation of this system with nine-year old children revealed that their ability to recognise and use appropriately the structural conventions embodied within a KREEK text varied with the mode they were in. They had little difficulty in understanding the fact that the positioning of the facts within the hierarchy provided additional information when they were *browsing* a KREEK text, but only the higher-ability children used these structural conventions appropriately in *authoring* mode (Purchase, 1992). Further experimentation needs to be done to supplement these observations with a more rigorous qualitative analysis, encompassing a larger sample size, and focusing more particularly on the structural envisagement aspect of text browsing and creation, and its support of the learning activity.

Depiction of 'lexical space'

GLOSS is currently under development: it is intended to be a language tool which portrays the highly-linked information in a thesaurus graphically, providing the means for easy and efficient access of the entries for the words cross-referenced by a particular lexical entry, and displaying these relationships between the entries in a graphical manner. Hypertext type implementations of thesauruses have implicit links between the entries: the system proposed here will make these relationships graphically explicit, thus allowing the user to visualise the 'lexical space' in which the word is situated. Thus the 'lexical area' defined by a word and its lexical associations will be depicted in a graphical manner (using a node-arc notation similar to Quillian's 'semantic

[1] This idea is similar to Quillian's linking together of planes (Quillian, 1967), without the condition that links between planes be restricted to relationships between concepts and their corresponding instances.

memory' representation (Quillian, 1967)), in order to allow the relationships between the words and their cross-references to be perceived visually. A lexical entry will be visually depicted as a 'focus' node representing the word, which is joined by graphical arcs (indicating lexical or semantic relationships) to the relevant cross-referenced words. The display is made more complex by the further arc-node depiction of words associated with these cross-references. User selection of any word which is displayed on the screen in one lexical entry would change the graphical display so that the newly selected word forms the focus of the display. The new focus node, its cross-referenced words, and further lexical or semantically associated words would now be displayed. By appropriate selection of words, the user can browse through the thesaurus, following relationships through the lexical space. A system like this may have many uses in education, in learning about the structure of the lexicon (either for children or for people learning English as a second language), or for discovering the semantic relationships between concepts. In addition, the graphical aspect of GLOSS provides any user with a new way of visualising words, semantic and lexical relationships and lexicon structure.

Schematic structures for interactive fiction

STORYTIME is a creative writing authoring tool, which allows non-sequential stories to be both read and written. It maintains a hypertext type implementation of events in a story, with the links between the events relating to choices that the reader makes when decision points are reached in the narrative. The non-sequential nature of the stories is what is of most interest from both a reading and an authoring perspective, and this underlying structure of highly-linked nodes allows many stories with different conclusions to be explored and created. STORYTIME is intended to be used by primary school children in both reading and authoring mode, and it provides a simple interface for both activities.

Readers of STORYTIME stories suffer from the problems that users of hypertext systems are prone to: not knowing where in the linked structure they are currently located, and having no prior knowledge of the scope of the story (in terms of both size and possible events in the narrative).[2] When creating stories, authors may experience the same problems, but these may be exemplified by errors or omissions that they may have made (*eg* forgetting to ensure that all pages are accessible, associating the wrong page with a link, or losing sight of the overall structure of the story, and the various narratives that it may entail.)

STORYTIME has been partially extended in an attempt to address these problems by including the facility for the overall structure of the story to be viewed as a schematic display of a directed graph. Thus the internal hypertext representation of nodes and links is made visible to assist the reader in knowing something more about the structure and entent of the story. In addition, the author will be able to get an overview of the possible paths in the story, and ensure that all pages are accessible and the created structure correctly represents their intention. Further implementation is necessary to provide graphical authoring facilities.

Like the thesaurus implementation discussed above, these extensions to STORYTIME require the use of a suitable automatic graphical display algorithm which will display the structure of the highly-linked internal representation of nodes and links. Graphical tools which implement algorithms which enable the perspicuous display of graphs (*eg* the 'spring' algorithm (Kamada, 1989)) will be used.

[2] A reader of a book may get some idea of its scope by seeing how thick it is, or by "flicking though" to get some preliminary notion of the nature of the language and events that can be expected. There is no similar equivalent in STORYTIME.

Conclusion

The three systems described above are intended to be used to investigate the potential for the use of schematic representations in an interactive device for three particular learning activities: browsing the representation of a structured domain, navigating lexical space, and accessing the structure of a story in an interactive fiction authoring tool. The aspect of these systems that distinguishes them from other computer educational systems which use schematic representations is the fact that what is represented is declarative subject matter, rather than the basis for a simulation tool (Cheng, 1993), or a procedural exercise (Levonen and Lesgold, 1993). Successful use of these systems (*ie* the enhancement of the respective learning activities) would demonstrate the suitability of this choice of medium, and would indicate that the interactive nature of the computer can indeed be exploited with schematic representations for the effective depiction of declarative subject matter.

Acknowledgements

I am grateful to Julie Stewart, Wendy Scherman, J. Mark Kennish, Kristin Smeltzer, Brent Robinson, Peterhouse, and the Cambridge Livingstone Trust.

References

BRUNER J.S.(1966). On cognitive growth. In BRUNER J.S., OLVER R.R., AND GREENFIELD P.M., editors, *Studies in Cognitive Growth.* Wiley.

CHENG P. C-H.(1993). Multiple interactive structural representation systems for education. In R. COX, editor, *Workshop on Graphical Representations, Reasoning and Communication, World Conference on Artificial Intelligence and Education, Edinburgh.*

DE SAUSSURE F.(1959). *A Course in General Linguistics.* Fontana.

KAMADA T.(1989). *Visualizing abstract objects and relations.* World Scientific.

LEVONEN J.J. AND LESGOLD A.M.(1993). Visual reasoning in coached practice environment: visual meaning making in SHERLOCK. In R. COX, editor, *Workshop on Graphical Representations, Reasoning and Communication, World Conference on Artificial Intelligence and Education, Edinburgh.*

MCDOUGALL A.(1990). Children, recursion and LOGO programming: An investigation of Papert's conjecture about the variability of Piagetian stages in computer-rich cultures. In *Proceedings of the 5th World conference on Computers in Education, Sydney.*

MINSKY M.(1985). A framework for representing knowledge. In BRACHMAN R.J. AND LEVESQUE H.J., editors, *Readings in Knowledge Representation.* Morgan Kaufmann.

OLSON D. AND BRUNER J.S.(1987). Learning through experience and learning through media. In BOYD-BARRETT O. AND BRAHAM P., editors, *Media, Knowledge and Power.* Croom Helm.

PAPERT S.(1980). *Mindstorms: Children, Computers, and Powerful Ideas.* The Harvester Press.

PEA R.D.(1987). LOGO programming and problem solving. In SCANLON E. AND O'SHEA T., editors, *Educational Computing.* Wiley and sons.

PEIRCE C.S.(1977). *Semiotic and significs: the correspondence between Charles S. Peirce and Victoria Lady Welby.* Indiana University Press. Edited by HARDWICK C.S.

PURCHASE H.C.(1992). The use of an educational computer system embodying a schematic method of knowledge representation. Technical Report F-INFENG/TR.110, Cambridge University Engineering Department.

QUILLIAN M R.(1967). Word concepts: A theory and simulation of some basic semantic capabilities. *Behavioral Science*, 12:410–430.

Automatic sound generation for spherical objects hitting straight beams based on physical models

M. RAUTERBERG
Usability Laboratory, Work and Organisational Psychology Unit
Swiss Federal Institute of Technology (ETH), Nelkenstrasse 11, CH-8092 Zurich, Switzerland

M. MOTAVALLI
Swiss Federal Laboratories for Material Testing and Research (EMPA)
Uberlandstrasse 129, CH-8600 Dubendorf, Switzerland

A. DARVISHI and H. SCHAUER
Department of Computer Science (Ifl)
University of Zurich, Winterthurerstrasse 190, CH-8057 Zurich, Switzerland

Abstract: The objective of this paper is the development of concepts, methods and a prototype for an audio frame work. This audio frame work shall describe sounds on a highly abstract semantic level. We describe every sound as the result of one or several interactions between one or several objects at a certain place and in a certain environment. The attributes of every interaction influence the generated sound. Simultaneously, the participating objects, which take part in the sound generation process, can consist of different physical conditions (states of aggregation), materials as well as their configurations. All relevant attributes have an influence on the generated sound. The hearing of sounds in everyday life is based on the perception of events and not on the perception of sounds as such. For this reason, everyday sounds are often described by the events they are based on. In this paper, a framework concept for the description of sounds is presented, in which sounds can be represented as auditory signal patterns along several descriptive dimensions of various objects interacting together in a certain environment. On the basis of the differentiation of purely physical and purely semantic descriptive dimensions, the automatic sound generation is discussed on the physical and semantic levels. Within the scope of this research project, we shall especially look for possibilities to describe the sound class 'solid objects', in particular the class of the primitive sounds 'knock' ('strike', 'hit'), because this class of sounds occurs very frequently in everyday life, the interacting objects can be easily and well described by their material characteristics and the knowledge of solid state physics can be used. As an example the falling of a spherical elastic object onto a linear elastic beam is physically and mathematically modelled, and implemented on a SGI workstation. The main parameters which influence the impact behaviour of such objects will be discussed. On the theoretical level, first a better overview and a better understanding of the capabilities, restrictions and problems of the existing instruments (tools) for the automatic generation of audio data can be anticipated.

State of the art

Many computers have a sound generator that produces a simple beep sound to indicate the errors. For a long time the only audio information for user interfaces contemporary modern workstations have signal processors and analogue digital converters and therefore sounds can be used in software. The terms that are used later in this chapter are defined as follows. *Audio signal pattern*: description of all perceptible audio signals. *Speech*: The description of all audio signals that have describable grammar structures. Music: Complex audio signal pattern that has rhythmic describable structures. *Tone*: Simple audio signal pattern with rhythmic describable structure. *Everyday sounds*: Audio signal patterns that have not been sufficiently researched to give a description and creation structure. We call the combined group of everyday sound and tone 'sound'. When we mean the assignment of a sound to an event we put the event identifier in inverted commas (for example: event = impact, sound = 'impact').

Comparison of auditory and visual signal pattern

The textual representation of information is of most use when the user is familiar with the domain area and can demonstrate much experience and knowledge in that domain area. In comparison, more concrete (visual and auditory) representations of information that the user can query are of most use when the domain area is new and unknown. By comparing audio signal patterns with visual signal patterns the different advantages of each can be shown. Sounds and music can be used to improve the user's understanding of visual predecessors or can stand alone as independent sources of information. (For example: sounds as diagnostic support applied with the direction of a process simulation (Gaver, 1991).)

The parallel use of different media and the resulting parallel distribution of information, for example by simultaneously showing a predecessor through a concrete representation and its explanation through audio distribution, leads to a denser sharing of information. In this case, the user can dedicate his attention solely to the visual information, which has parallel audio support. This reduces the need to change the textual or other visual delivery and prevents the overflow of visual information (Gaver, 1991). The redundancy of information represented visually and auditory, as long as the representation of the information is realistically formed, is sensed not as disturbing, but instead it demands and increases information reception. It is important that with simultaneous information representation, that the information is harmonised together and that the different media are well synchronised.

Everyday sound perception

The perception of auditory signal patterns in everyday life can come in very different forms: a car driving by on the street, a dripping faucet, the confusion of voices from a crowd of people, opera music, a plane flying by, the buzz of a travel alarm clock, the beeping of a wristwatch, etc. All of these auditory signal patterns are divided into four categories: speech, music, sound and noise; sometimes noises and sounds are heard and grouped together. All of these categories are described sufficiently in the physical world through the mixing and superposition of different pitches, frequencies, volumes and sound duration. One of the essential differences between these categories, however, lies in their semantics: Speech serves primarily to convey information, while music and noise can have a pleasant or an unpleasant influence on the emotions. For musicians and other people who are intimate with this area, music and noise have a comparable semantic and informative character as speech does for the normal citizen. Besides from music and noise, the listener is interested next in the possibility of undisturbed, context free perception.

In contrast to music and noise, everyday sounds have a self standing characteristic; they are extremely context sensitive and event related (Gaver, 1986). Through the physical interaction of different everyday objects in 3-D space the sounds of everyday life are created and through propagation they become audible through the air. In comparison with music and noises, the semantically relevant dimension of sound lies not with the characteristic quality of the auditory signal pattern itself, but rather with the quality of the sound producing event as it respects the concerned object (Mountford, 1990).

This difference leads us to the conclusion that sounds are interpreted differently than music based upon their quality. When listening to music we are primarily interested in the effect of music on us; while when hearing everyday sounds we are interested in the quality of the sound producing object and the accompanying circumstances (e.g., surrounding conditions, events, etc.). Of course, music can be heard from the perspective of every day use; in this case the listener pays attention to the nature and tune of the instrument in use, to the tempo, to the acoustic, to the place of performance, etc. This method of listening to music is dependent upon the listener's knowledge of this domain field; only someone who is experienced with music will be able to extract all the various aspects from a piece of music. The average adult is, for the interpretation of everyday sounds, an expert with a large degree of knowledge from experience. This knowledge allows one to evaluate everyday sounds according to the following criteria for relevant information:

1) Information about the physical occurrence: we hear, if the fallen glass clinks or breaks.
2) Information about unseen structures: when knocking on the wall, we can hear if it is hollow.
3) Information about dynamic changes: when filling a wine glass, we can hear when it is full and runs over.
4) Information about abnormal conditions: we hear, when the car engine ceases to function properly and runs irregularly.
5) Information about occurrences outside of the visual field: the sound of footsteps behind us 'tells' us if someone is approaching.

Listening to everyday sounds is based upon the perception of events and not upon the perception of sounds in and of themselves. This fact becomes clear in the following example:

Illustrative example:
A pen dropped upon a piece of paper from a height of about 15 cm created a different sound than when it is dropped upon the hard surface of a desk. An altogether different sound is created when a rubber eraser is dropped upon the paper or, respectively, on the desk.

The sound created in each case of the previous example is neither a characteristic of any of the participating objects (pen, rubber eraser, sheet of paper, desk surface) nor a characteristic of the occurrence 'dropped' itself. The four different sounds in the examples are, with an observation that holds true to the reality of the situation solely determined by their respective interaction and environmental conditions. Everyday sounds are therefore due to a lack of better descriptive possibilities, often described through the underlying occurrence.

Every sound is also a result of one or more interactions between two or more objects in a definite place and in definite surroundings and can be defined as the following:

$$\text{Sound} = f\ (\text{objects, interaction, environment})$$

Every interaction possesses attributes that have an influence on the produced sound. At the same time the shared objects can participate in the production of sound from different aggregate conditions, materials, and even their configuration. The configurations of these materials possess attributes that also can have an influence on the produced sound.

Existing research projects

With the manufacture and adaptation in the world of interactive computer systems (e.g., multimedia applications) one must decide between the following three types of substitution: 1) *Propagation*: Here the quality of the propagation channel is important (i.e., bandwidth, etc.). 2) *Intake, saving, and reproduction*: this is actually time synchronous with visual processing to add extra significance. 3) *Automatic Production*: In this area the realistic generation of context sensitive, auditory signal pattern has the most significance.

We concentrate us in our research to the third area: the automatic production of every day sound. The existing research for audio with interactive systems will be listed. The computer manufacturer, Apple, researches the substitution of audio media for improvement of graphical user interface and presents the so called 'Auditory Icons' such as 'Earcons' (Gaver, 1986; Gaver, 1989; Gaver, 1990; Blattner, 1989). The Bell Communications Research Centre in collaboration with the Massachusetts Institute of Technology developed an 'Audio Window system' as an analogue of the 'Visual Window system' that realised one, two, and three dimensional representations of auditory source (Ludwig, 1990; Wenzel, 1991). The Olivetti Research Centre was developing a so called 'Audio Server' an analogy to the 'Window Server' with a user interface based on the Client Server Model (Binding, 1990). None of the three projects mentioned above, however, allow the description of sounds on a highly abstract semantic level.

State of own research

It is mainly the following factors in their respective forms and combinations that influence the load for humans in the man computer interaction: the organisation of the learning place and the learning place environment, the content of the learning tasks on the screen, the temporal share of the activity on the screen in proportion to the whole activity, the content of the computer support and the user friendliness of the interactive system.

Our investigations have shown that the user friendliness of a data processing system crucially influences its acceptance by its users. Three different aspects should hereby be discriminated: (1) the functionality and the amount of information, (2) the availability (response times and disturbances) and (3) the operational handling mediated by the user interface (Rauterberg, 1992).

The results of an experiment, that investigated the effects of audio feedback, showed, that the results of a database query at the user interface with an individually selective acoustic feedback were as good as with a previously adjusted standard interface without any acoustic feedback. However, if the users, who considered the acoustic feedback as useful, are compared with those, who considered it not necessary, significant performance advantage results for the persons with a positive mind about the acoustic feedback (Rauterberg et al, 1991).

The results of a second experiment with or without sound feedback in a process control environment showed, that sound feedback improves significantly the performance of operating and controlling the simulation system of an assembly line with 38 different sounds (Rauterberg & Styger, 1994).

Definition of the main research goal

From the analysis of existing models and concepts as well as empirical investigations of the pre phase a definition for the requirements for a new concept and a new model should be derived. The example of an impact (e.g., falling spherical object on the surface of a beam) shows the physical process of producing sound through the interaction of the objects 'spherical object' and 'beam' (fig. 1). The produced sound 'spherical object hits the beam' can be described by the behaviour of the vibration of the beam surface, which has been produced by the deformation resulting from the fall of the spherical object. "When an object is deformed by an external force, internal restoring forces cause a build up potential energy. When the external force is removed, the object's potential energy is transformed to kinetic energy, and it swings through its original position. The object continues to vibrate until the initial input of energy is lost" (Gaver, 1993).

The following concept should help us to find a classification to describe the sounds on a very high abstract and semantic level. The highest level in our sound hierarchy contains the class of everyday sounds that are produced through interaction of two or more objects (interacting objects). The second highest level contains the three different material states (solid, liquid, gaseous), which produce qualitatively diverse sounds. For example, if two or more solid objects interact, then they produce sounds like 'push', 'break', etc. Sounds like 'drop', 'sprinkle' are produced through interaction of different liquid objects, and sounds like 'explode', 'stream' are produced through aerodynamic interactions in the air. All of these so called 'primitive sounds' are characteristically for their classes.

Audio framework

The term 'framework' comes from the object oriented world and is understood as follows: Already programmed frames that allow the programmers to supplement the application of specific parts in a program. An audio framework should in the term of object-oriented terminology provide the basic sound classes and the possibilities to combine and manipulate them. At the moment there is no such audio framework. As an example we are going to approach the sound class 'impacts'.

Our methodology can be demonstrated by falling a linear elastic spherical object with the elasticity modules E_1, Poisson ratio ν_1, mass m and radius r_s onto a straight beam (fig. 1). The material behaviour of the beam is assumed to be isotropic and linear elastic with the elastic modules E, Poisson ratio ν, cross sectional area A = (h * b), length l and the material density ρ. First, we will ignore the damping of the system which is mainly affected by the viscoelasticity effects of the material. The behaviour of anisotropic materials such as multi-layered composite structures has been previously investigated (see for example Motavalli, 1991). The results of this investigation can be used for the further developments.

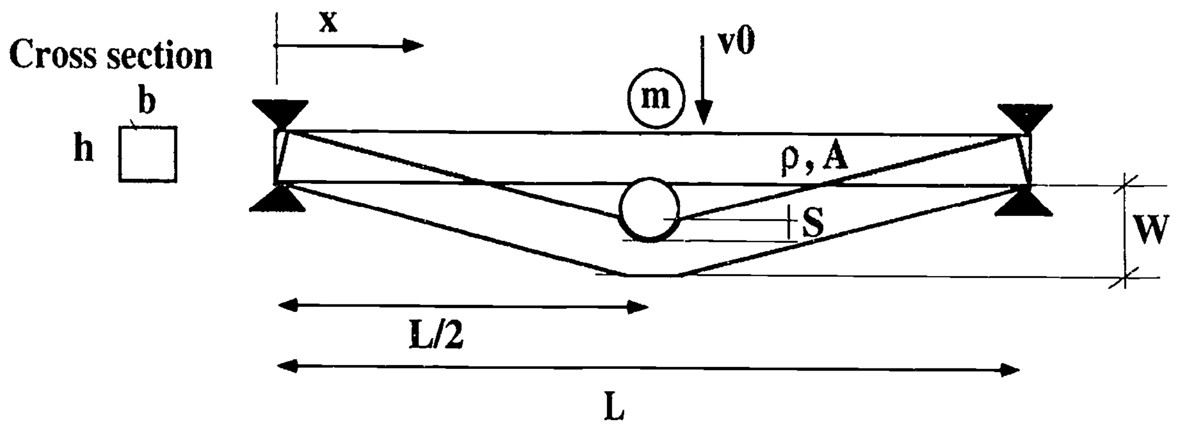


Fig. 1: Falling a mass on a straight beam

The displacement Z(t) of the centre of mass of the spherical object hitting the structure satisfies the equation of motion:

$$Z_{,tt} = -\frac{F}{m} \qquad (1)$$

where $(.)_{,tt} = d^2(.) / dt^2$ is the second time derivative, $F \geq 0$ is the resultant force acting at the contact region between the beam and the mass.

The resultant contact Force F can be connected with the maximum relative displacement S of the upper surface of the structure with respect to its middle surface according to (fig. 1) by solving the corresponding classical contact problem of linear elasticity (Hertz, 1881):

$$F = kS^{\frac{3}{2}} \qquad (2)$$

where $k = \frac{4}{3}\sqrt{r_s}\left(\frac{E' E_1'}{E' + E_1'}\right)$, $E' = \frac{E}{1-\nu^2}$ and $E_1' = \frac{E_1}{1-\nu_1^2}$.

One of the most essential assumption in deriving equations (1) and (2) is that the elastodynamic effects in the region of contact are neglected. Furthermore the accuracy of the following equations is quite satisfactory as long as the radius r_S of the sphere is sufficiently large with respect to the beam thickness h (for example, $r_S/h \geq 2$).

The differential equation of the motion of the beam under distributed load q(t,x) in terms of lateral displacement W is:

$$EIW_{,xxxx} + \rho A W_{,tt} = q(t,x) \qquad (3)$$

where $(.)_{,xxxx} = d^4(.) / dx^4$ and $I = (bh^3)/12$.

Taking the boundary conditions into account, this equation can be solved analytically. After some mathematical operations, the following solution can be derived:

$$W(t,x) = \sum_n \frac{2}{\rho AL}(-1)^{\frac{n-1}{2}} \sin\left(n\frac{\pi x}{L}\right)\left[\frac{1}{\omega_n}\int_0^t F(\tau)\sin(\omega_n(t-\tau))d\tau\right] \qquad (4)$$

The displacement Z(t) of the centre of the sphere will be given by the sum of the two displacements S and W (fig. 1):

$$Z = S + W \qquad (5)$$

The displacement W at the mid span of the beam is:

$$W(t,L/2) = \sum_{n=1,3,5,\ldots} \frac{2}{\rho AL\omega_n}\int_0^t F(\tau)\sin(\omega_n(t-\tau))d\tau] \qquad (6)$$

where ω_n are the eigen frequencies of the beam.

Provided that one considers the material damping as an additional effect to the above mentioned model, the following equation can be derived instead of (4):

$$W(t,L/2) = \sum_{n=1,3,5,\ldots} \exp(-\delta_n t)\frac{2}{\rho AL\omega_n}\int_0^t F(\tau)\sin(\omega_n(t-\tau))d\tau] \qquad (7)$$

where δ_n is the damping constant of the beam material.

Equations (2), (5) and (7) illustrate the dependency of the beam-surface vibration (S + W) on the following important parameter:

- material properties and geometry of the beam and
- material properties, geometry and initial velocity of the falling object.

The main advantage of the presented physical model to the existing model in (Gaver,1993) is that the material and geometrical properties of both objects are considered rather than the properties of the impacted beam only. Further investigations are needed to develop solutions for objects with other geometry.

The sound 'ball hits the straight beam' can be automatically generated using the appropriate parameters derived from the physical model of the interacting objects. Using a flat plate (Koller, 1983) and/or a non flat plate (Sayir, 1992) can cause complex mechanical equations.

The importance of an audio framework

Modern workstations (e.g., NEXT, SGI) provides sound and music kits. They also provide so called basic operations for manipulation of sound patterns in the frequency level. In addition they offer the opportunity to assign a digitised sound to a certain dialogue element, for example a button. All of these approaches assign an unchangeable sound to an object or operation. These sounds are very synthetic. It means context free and does not fulfil our expectation of the real world. Instead of that we understand a sound as a result of one or more interactions between two or more objects in a certain place and a certain environment. It means that a sound can convey more information about the sound source, it's place and environment. Therefore it should be calculated in real time and is context sensitive.

On the theoretical level a better overview and understanding of the skills and restrictions of existing tools for automatic generation of audio data are expected. The new concepts that are planned for development will give a basic impulse to the following fields. *Multimedia*: New possibilities for interactive creation of sound producing events and actions. *Simulation* , *Computer supported learning*. A pedagogic target of computer support learning will clarify the processes. These processes could be physical or chemical processes. *User interfaces* (e.g., for the visually impaired): Simulations also find their applications in the future for handicapped and especially visually impaired computer users. For example, blind computer users could recognise the dialogue objects and movable pictures in a 3-D space by sounds. *CAD, Architecture*: New possibility of examining the sound isolation of rooms, walls, materials, etc. Generally, the acoustics of buildings could be changed and checked interactively.

References

Binding, C., Schmandt, C., Lantz, K. & Arons, B. (1990) Workstation audio and window-based graphics: similarities and differences. in G. Cockton (ed.) Engineering for Human-Computer Interaction. (pp. 233-247), Amsterdam: North-Holland.

Blattner, M. et al. (1989) Earcons and Icons: their structure and common design principles. *Human Computer Interaction*, 4:11-44.

Gaver, W. (1986) Auditory Icons: using sound in computer interfaces. *Human Computer Interaction*, 2, 167-177.

Gaver, W. (1989) The SonicFinder: An Interface That Uses Auditory Icons. *Human Computer Interaction*, 4, 67-94.

Gaver, W, & Smith, R. (1990) Auditory icons in large-scale collaborative environments. in D. Diaper et al. (eds.) Human Computer Interaction INTERACT'90 (pp. 735-740). Amsterdam: Elsevier.

Gaver, W. (1991) The SonicFinder: An Interface That Uses Auditory Icons. *Human Computer Interaction*, 4, 67-94.

Gaver, W., Smith, R. & O'Shea, T. (1991) Effective sounds in complex systems: the ARKola simulation. in S. Robertson, G. Olson & J. Olson (eds.), Reaching through technology CHI'91. (pp. 85-90), Reading MA: Addison-Wesley.

Gaver, W. (1993) What in the World do We Hear? An Ecological Approach to Auditory Event Perception. *Ecological Psychology*, 5(1).

Hertz, H. (1881) Über die Berührung fester elastischer Körper, Journal für die reine und angewandte Mathematik, vol. 92, pp 156-171.

Koller, M. G. (1983) Elastischer Stoss von Kugeln auf dicke Platten. Dissertation Nr. 7299, Zuerich: ETH.

Ludwig L., Pincever, N. & Cohen, M. (1990) Extending the notion of a window system to audio. *Computer*, 23(8), 66-72.

Motavalli M., (1991) Das Verhalten mehrschichtiger Verbundrohre unter Innendruck. Dissertation Nr. 9597, Zuerich: ETH.

Mountford, S. & Gaver, W. (1990) Talking and listening to computers. in B. Laurel & S. Mountford (eds.) The Art of Human-Computer Interface Design. (pp. 319-334), Reading MA: Addison-Wesley.

Rauterberg, M., Baumgartner, A., Jenny, A. & de Lange, M. (1991) Erhoehung der Transparenz durch individualisierte Soundunterstuetzung bei Datenbankrecherchen. Technical Report IS-1-91. Institut fuer Arbeitspsychologie. Zuerich: Eidgenoessische Technische Hochschule.

Rauterberg, M. (1992) An empirical comparison of menu-selection (CUI) and desktop (GUI) computer programs carried out by beginners and experts. *Behaviour and Information Technology*, 11(4), 227-236.

Rauterberg, M. & Styger, E. (1994) Positive effects of sound feedback during the operation of a plant simulator. submitted to "East-West International Conference on Human-Computer Interaction EW-HCI'94" in St. Petersburg (Russia), August 2-6, 1994.

Sayir, M. B., Haeusler, K. & Partel, M. (1992) Impact Behavior of corrugated sheets of fibre-reinforced cement. *Composites*, 23(6).

Smith, S., Bergeron R. & Grinstein, G. (1990) Stereophonic and surface sound for exploratory data analysis. in Proceedings of Computer-Human-Interaction CHI'90. New York: ACM.

Wenzel E., Wightman F. & Kistler D. (1991) Localization with non-individualized virtual acoustic display cues. in S. Robertson, G. Olson & J. Olson (eds.), Reaching through technology CHI'91. (pp. 351-359), Reading MA: Addison-Wesley.

A Methodology for Analyzing Students' Interactions within Educational Hypertext

Margaret M. Recker
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332-0280 U.S.A.
email: mimi@cc.gatech.edu

Abstract: We present a theoretical approach and a methodology for analyzing data from students interacting with and learning from hypermedia systems. In our approach, interactions are viewed to be mutually influenced by individual students' goals and strategies and the actions supported by the interface of the learning environment. The approach is illustrated by modelling data from an empirical study in which students browsed through a hypertext instructional environment to learn about programming concepts. By using the explanatory power of the computational model, interactions can be analyzed to determine patterns of use. Results obtained from this method of analysis yield specific feedback on system design and prescriptions for improving the design. More theoretically, they provide valuable insights on the nature of human cognition and learning in the context of interactive educational technologies.

Designers of hypermedia systems can never completely anticipate how people will use their systems. This is especially true in educational settings where, typically, users are both domain and hypermedia novices. In these situations, learners lack both a mental model of the domain represented in the system and effective browsing strategies. Therefore, empirical evaluations of system design become paramount. Ideally, such evaluations will provide concrete prescriptions for improving the design. Unfortunately, the identifiability problem is often difficult. Even a cursory review of evaluations of educational technologies reveals a cloudy, complex picture. Empirical results have shown that these learning systems are often used in ways that are completely unintended by its designers, are unproductive, result in aptitude-treatment interactions, and show large individual differences (Jonassen and Mandl, 1990; Steinberg, 1989). Compounding this problem is the fact that it is often difficult to attribute empirical results back to specific features in the design.

In this paper, we present a theoretical approach and a methodology for analyzing student data as they interact with and learn from hypermedia systems. In our approach, interactions are viewed to be mutually influenced by individual students' goals and strategies and the action opportunities supported by the interface of the learning environment (Kirlik, 1993; Pirolli and Wilson, 1992). In our method, which draws upon techniques from both artificial intelligence and the cognitive sciences, student interactions are modelled in terms of both student goals and environmental constraints on action. By using the explanatory power of the computational models, interactions can be analyzed to determine patterns of use. Results obtained from this method of analysis yield specific feedback on system design and prescriptions for improving the design. More theoretically, they provide valuable insights on the nature of human cognition and learning in the context of interactive educational technologies.

The methodology is illustrated by using data from an empirical study in which students browsed through a hypertext instructional environment to learn about recursive programming concepts in Lisp. Following our method, the data are modelled by coupling simulations of individual students' actions with a model of action opportunities supported by the interfaces of the instructional environment. The resulting simulations capture the interactions observed in the empirical data. Specifically, for each student, the corresponding simulation is required to perform students' mouse actions and their verbal utterances, in exactly the same temporal order that these were observed in the data.

The simulations of individual students are analyzed using a statistical clustering algorithms in order to identify common usage patterns across students. These patterns, we argue, represent students' strategies for navigating and learning within the hypertext environment. The explanatory power of our cognitive model provides the semantics for interpreting these patterns in terms of successful and unsuccessful design features of the hypertext system. In addition, it yields specific prescriptions for improving the design.

The remainder of this paper is organized as follows. In the next section, we briefly describe the hypertext learning system and its empirical evaluation. We then present our modelling methodology, and show how the model is used to identify patterns of learner use. We close with a discussion of how an understanding of these patterns can help inform system re-design.

System Description and Empirical Evaluation

Elsewhere we have described in detail the design of a hypertext learning environment, called the Explanation Environment, which contains instructional materials (text and examples) on programming recursion in Lisp (Recker and Pirolli, in press). Briefly, in the Explanation Environment, instructions can be browsed in a non-linear fashion. The environment also contains instructional examples, which are annotated with explanatory elaborations that students can choose to view by clicking with on a button. With this design feature, we hope to provide extra, optional explanations to students who are unable to produce them on their own; students who are able to generate their own explanations for the examples can choose to ignore these additional textual elaborations.

Students are provided with two navigational methods for moving between top-level screens of the instruction. The first, *global navigation*, provides learners with a map of top-level topics, each listed on a button. The second navigational method, *local navigation*, is implemented by providing two buttons on each instructional screen, which learners can click on to move to the next and previous top-level instructional topic, respectively.

An empirical study was conducted in order to examine the effects of the hypertext system on students' initial understanding and their subsequent performance while programming recursion. In this study, students went through five lessons on programming in Lisp. Each lesson had two parts: studying instructional material (learning), followed by programming (problem solving), using an intelligent tutoring system for Lisp, the CMU Lisp Tutor (Anderson et al., 1990). For the target lesson, the lesson on recursion, two sets of computer-based instructions were developed. Students were randomly assigned to one of the two environments to learn about the concepts of recursion prior to programming recursion with the Lisp Tutor. The first environment was the Explanation Environment. A second, control instructional environment was also implemented, which mirrored more standard, linearly structured instruction. Students navigated through both environments by clicking on buttons. These mouse actions provided additional data on students' strategies for learning from instruction.

When we contrasted subjects' performance while programming with the Tutor, we did not find any significant differences in outcome between subjects using the hypertext-based instructional system and those in the control. However, we did find a significant aptitude-treatment interaction. Post-hoc analyses showed that the higher-ability subjects (those that performed well in the pre-intervention lessons) in the hypertext condition made significantly less errors while programming than the low-ability subjects. Subjects in the control condition did not show significant ability-based differences. The full results of the study are presented elsewhere (Recker and Pirolli, in press). In this paper, we focus on the models of student interaction in the hypertext condition and show how we analyzed these to identify common patterns of use.

Modelling Technique

Data from the empirical study form the motivation for a computational model, called SURF (Strategies for Understanding Recursive Functions). SURF is implemented within the Soar architecture (Laird et al., 1987). Soar is an AI production system architecture in which problem solving is carried out by search through problem space in order to achieve particular goals. Soar also includes an experience-based learning mechanism, called *chunking*, which summarizes problem solving experiences into a more efficient form.

In the interest of space, we can only present an overview of the SURF model. More details can be found elsewhere (Recker and Pirolli, in press). The primary goal of the SURF model is to model the learning behavior of individual students in terms of two criteria. The first criterion requires that *every* mouse clicking action by all students is simulated in the exact order that it occurs in the data. This forms what is called the *fine* modelling criterion. The verbal explanations that students made to themselves (they were asked to provide concurrent verbal protocols), which we call self-explanations (Chi et al., 1989; Pirolli and Recker, in press), form the secondary, *coarse* modelling criterion. This meant that students' self-explanations are modelled at a rough level, in the sense that their exact natural language statements are not simulated. More specifically, at each screen in the instructional environments, a student can attempt to self-explain the instruction. The students' verbal protocols are consulted to determine when such self-explanations are exhibited. At each of these instances, the corresponding simulation applies what is called the *comprehension* operator. The application of this operator results in the creation of chunks, representing newly acquired knowledge. In sum, the Surf model, focuses on exactly capturing the temporal sequence of students' interface interactions and their self-explanations.

In the SURF model, student-environment interactions are modelled as two components: (1) a simulation of the possible actions support by the instructional interface, and (2) simulations of individual students' interaction and learning strategies, and their prior knowledge (jointly called *capabilities*).

The interface of the instructional environment is a Lisp simulation of the buttons and instructions that are displayed in each screen context. These buttons and instruction snippets represent opportunities for actions. The presence of buttons offer mouse-clicking opportunities, while instructional snippets present self-explanation opportunities.

Learners' Capabilities. The second component of the model simulates individual learners' capabilities. In Soar, a set of production rules is created for each student, called the learner's *profile*, which represents each student's learning strategies and prior knowledge. Each student's profile is implemented such that when it is loaded in with the interface model, the resulting Soar run fits that student's behavior. Recall that the fit has to meet two criteria: the *fine* criterion requires that every mouse clicking action is captured occurs and the *coarse* criterion models students' self-explanations at a rough grain-size.

Two kinds of methods are currently implemented for modelling learners' capabilities. First, a set of production rules represents the learner's prior knowledge that is used to generate a self-explanation. A second set of production rules represents how the learner selects among possible available actions (or operators). In Soar, the desirability or acceptability of possible alternatives is described in terms of a fixed language of *preferences* (e.g., best, better, reject). In SURF, preference productions are used to express the value of available operators (e.g., selecting a particular button is desirable) in order to simulate a student's actions in the order that these occur. Since preference productions deliberately choose among available operators (and thus are knowledge about knowledge), they can be seen as representing strategic or metacognitive knowledge.

Model Analysis

The preference production rules in a student's simulations models that student's mouse clicking actions and (roughly) verbal explanations, in the exact temporal order that these occurred. These preference productions therefore represent a student's strategic knowledge for explicitly choosing among available actions. In short, they are tangible representations of learning and navigating strategies.

The profiles can be analyzed in order to identify patterns of common use across students. In order to accomplish this, we constructed a dichotomous correlation matrix of all preference production rules that occurred more than once in the union of all student profiles. The resulting matrix of 26 production rules is used to perform a multidimensional scaling, using two dimensions.

As can be seen in the resulting plot Figure 1, several clusters of related of production rules, shown as letters, are evident. These clusters can be interpreted by using the semantics of the cognitive model

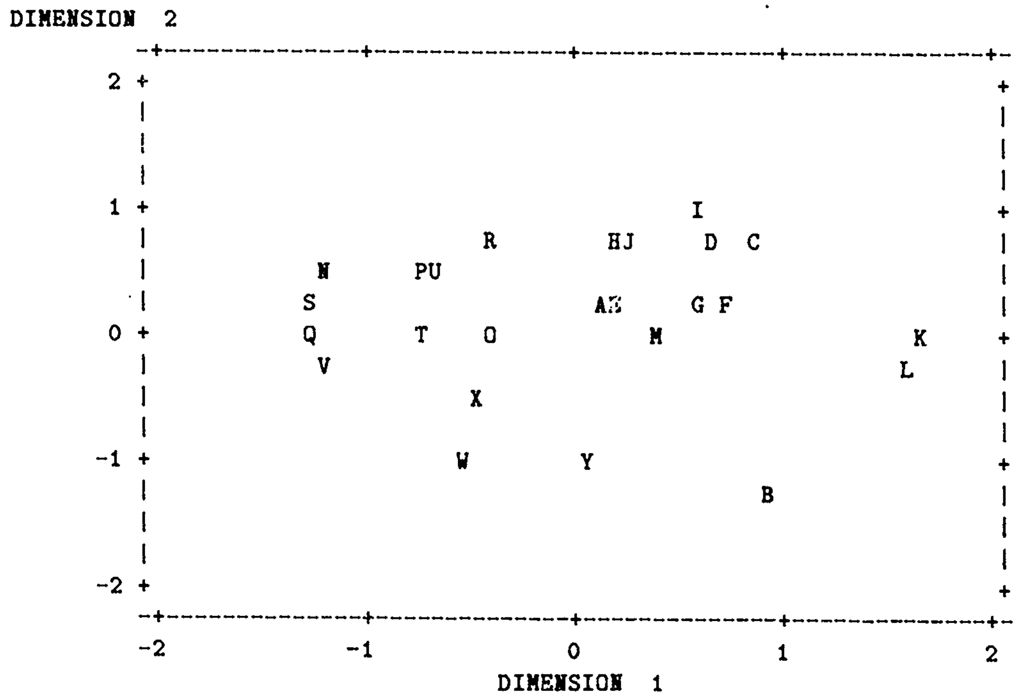


Figure 1: Multi-dimension scaling of preference production rules in students' profiles

(expressed in terms of interface primitives), as follows:

- Group A, E. This cluster contains production rules for navigation using the top-level "Map of Topics" global navigation facility.
- Group H, J. This cluster contains production rules that express a preference for viewing the instructional examples, when available.
- Group D, C, I. This cluster contains production rules that express a preference for a forward, serial navigation strategy and for avoiding backtracking.
- Group F, G. Similar to the second cluster, this cluster contains production rules that express a preference for *studying* instructional examples, especially in lieu of textual instructions.
- Group U, P, T. This cluster contains production rules that involve screen management, specifically for keeping track of screens that are skipped. Since these productions keep track of screens, they make high working memory demands and, as such, probably impose an added cognitive demand.
- Group N, S, Q, V. This cluster is similar to the previous cluster (and occupied the same general area). These production rules keep track of screens that have been viewed and guard against viewing them again. As in the previous cluster, these production rules make high working memory demands.
- Group K, L. These two production rules appear unrelated and it remains unclear why they are clustered.

Implications for Design

Based on this analysis, several conclusions can be drawn about specific features of the hypertext system. Some of these conclusions pertain to the design of educational technology and some apply to understanding how students learn in the context of educational technologies.

First, the existence of a distinct global navigation cluster suggests that the tool is an important and relied-upon facility in the hypertext system. Furthermore, the ability to use this facility appears to be a separate, identifiable cognitive skill. Taken together, these imply the importance of providing high-level navigational aids within hypermedia.

Second, as a default browsing strategy, novices appear to prefer forward, serial browsing and to dislike backtracking. Such preferences have been found in other evaluative studies of hypermedia systems (Mayes et al., 1990), and may reflect a default, novice navigation strategy. Designers should ensure this default strategy is close to optimal when designing educational hypermedia. Additionally, designers ought to consider scaffolding methods to support and encourage the development of alternate browsing strategies.

Third, instructional examples appear to be highly valued by students. Students both preferred to select instructional screens containing examples and to study these examples. In fact, the reliance by novices on examples in the early phases of learning a new domain is a robust finding in the literature (LeFevre and Dixon, 1986; Pirolli and Anderson, 1985; Ross, 1984; Sweller and Cooper, 1985).

Fourth, the clustering of production rules that keep track of visited screens suggests that the hypertext environment does not possess adequate features for helping students mark their current location within the system, mark where they've been, mark what they've chosen to skip and determine what remains to be seen. As a result, learners have to do much of this bookkeeping for themselves, which possibly adds undue cognitive load and may interfere with learning. This highlights the importance of designing hypermedia interfaces that they also function as external memory aids.

Finally, what are we to make of singleton production rules (those that did not cluster) and the uninterpretable cluster? The singletons represent isolated behaviors, reflecting the myriad of ways that students can choose to interact with systems. The uninterpretable clusters represent patterns that are not accounted for by the interface. In short, they are unexpected behaviors. In general, it is unlikely that system design will ever completely eliminate these. We can only hope to minimize their occurrences, relative to the explainable patterns.

Discussion

In this paper, we have presented a framework and methodology for analyzing students' interactions and learning in the context of using a educational hypertext system. The framework posits interaction to be mutually influenced by environmental constraints (i.e., actions supported by the interface) and individual learner goals and strategies. The methodology, which relies on an analysis of a model of observed interaction, yields specific results for evaluating system design. These results also serve as prescriptions for improving the design.

Clearly, before claiming generality, the overall approach should be applied to other data of students interacting with educational technology. In fact, we are currently planning to use our method with data collected as students learned to troubleshoot a complex system through using a simulation coupled with an intelligent tutoring system, called Turbinia-Vyasa (Vasandani and Govindaraj, 1993).

In closing, it could be argued that this is an expensive methodology, since it requires the formulation and implementation of computational cognitive models. However, we note that the use of Soar is not a prerequisite. Any cognitive architecture in which units of skill and interface primitives can be represented will suffice. In addition, the time to implement the computational models should decrease with growing expertise. However, in closing, we argue that it will never be possible to completely obviate the need for a deep of understanding of student characteristics and interface capabilities in order to appreciate the myriad of ways students can adapt to educational technology.

References

Anderson, J., Boyle, C., Corbett, A., and Lewis, M. (1990). Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42(1):7–49.

Chi, M., Bassok, M., Lewis, M., Reimann, P., and Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, 13:145–182.

Jonassen, D. and Mandl, H., editors (1990). *Designing Hypermedia for Learning.* Springer Verlag, Berlin.

Kirlik, A. (1993). Modeling strategic behavior in human-automation interaction: Why and "aid" can (and should) go unused. *Human Factors*, 35(2):221–242.

Laird, J., Rosenbloom, P., and Newell, A. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence*, 33:1–64.

LeFevre, J. and Dixon, P. (1986). Do written instructions need examples? *Cognition and Instruction*, 3:1-30.

Mayes, T., Kibby, M., and Anderson, T. (1990). Learning about learning from hypertext. In Jonassen, D. and Mandl, H., editors, *Designing Hypermedia for Learning*, pages 227–250. Springer Verlag, Berlin.

Pirolli, P. and Anderson, J. (1985). The role of learning from examples in the acquisition of recursive programming skills. *Canadian Journal of Psychology*, 39(2):240–272.

Pirolli, P. and Recker, M. (in press). Learning strategies and transfer in the domain of programming. *Cognition and Instruction.*

Pirolli, P. and Wilson, M. (1992). Measuring learning strategies and understanding: A research framework. In *Proceedings of the International Conference on Intelligent Tutoring systems*, Berlin. Springer Verlag Lecture Notes in Computer Science.

Recker, M. and Pirolli, P. (in press). Modelling individual differences in students' learning strategies. *Journal of the Learning Sciences.*

Ross, B. (1984). Remindings and their effects in learning a cognitive skill. *Cognitive Psychology*, 16:371–416.

Steinberg, E. (1989). Cognition and learner control: A literature review, 1977-1988. *Journal of Computer-Based Education*, 16(4):117–121.

Sweller, J. and Cooper, G. (1985). The use of worked examples as a substitute for problem solving in learning algebra. *Cognition and Instruction*, 7:1–39.

Vasandani, V. and Govindaraj, T. (1993). Knowledge structures for a computer-based training aid for troubleshooting a complex system. In Towne, D., de Jong, T., and Spada, H., editors, *The Use of Computer Models for Explication, Analysis, and Experiential Learning.* NATO ASI Series F, Springer Verlag, Heidelberg.

Acknowledgements

This research was supported by NSF under contract IRI-9001233 to Peter Pirolli and by a University of California Regents' Dissertation Fellowship. Preparation of this manuscript was supported by the Army Research Institute under contract No. MDA-903-90-K-0112 to Janet Kolodner.

Composing with Images: A Study of High School Video Producers

BRIAN REILLY
Division of Language and Literacy, School of Education
University of California, Berkeley, Berkeley, CA 94720, USA

So I bought a .44 magnum it was solid steel cast
And in the blessed name of Elvis well I just let it blast
'Til my TV lay in pieces there at my feet
And they busted me for disturbing the almighty peace
Judge said "What you got in your defense son?"
"Fifty-seven channels and nothin' on"
— Bruce Springsteen,
"57 Channels (And Nothin' On)"

Abstract

Television is the dominant cultural and educational medium of the second half of the twentieth century, and although it has been regarded by many as a largely negative influence on society (c.f. Mander, 1991, Postman, 1985), the impact and reach of television continues to expand. More than half of the homes in the United States subscribe to cable television, and about 65% own videocassette recorders (Dorr, 1990). Television, as an educational medium in schools, has been around for 40 years (Cuban, 1986; Koenig, 1969), primarily as an information delivery platform.

At Bell High School, students have been using video cameras, computers, and editing machines to create videos in a variety of forms and on a variety of topics for more than five years. In this setting, video is the textual medium of expression. I will discuss how the "reading" and "writing" of video texts by these students is an example of a new form of literacy, one which combines print, video, and computer technologies and one where composers work collaboratively to produce texts which are easily shared with an audience.

Context

Bell High School is located in the city of Bell, in southeastern Los Angeles county, and is part of the Los Angeles Unified School District (LAUSD). The LAUSD serves over 630,000 students and is the second largest school district in the United States. Bell High School is a year-round school with over 4200 students on three tracks. 96% of the students at Bell are Latino, and 98% qualify for the Federal student lunch program. With a published dropout rate of 40% and an environment where drugs, gangs, and violence are commonplace, Bell is like many urban schools in the United States. And like many schools, it's also not the kind of place where one would expect to find students using advanced video and computer technology. Yet over the past four years, Bell students have won more than 100 awards for their work in local, state, national, and international video competitions. Student productions include public service announcements, video essays, video poems, and music videos covering a wide range of topics such as drugs, homelessness, gangs, teenage pregnancy, AIDS, and the LA civil unrest.

The video production program is part of the English department and is taught by two teachers, Ed Murphy and Larry Stone. Ed Murphy began working with film and video about 20 years ago and now teaches 3 television production classes in addition to 2 English composition/literature classes. Larry Stone teaches 3 English classes and 2 video production classes. During the '92–'93 school year, the two teachers also offered an after school video class 4 days a week, and during the week and over weekends they loan out cameras to students.

Methods

I spent the 1992-93 school year working with students and teachers at Bell High School to understand how and what they learned about in their use of video. I watched as they scripted, taped, directed, edited, and premiered their work, and I came to understand how the learning environment facilitated by their teachers combined a sense of community, collaboration, high standards, authentic assessment, and technology in a way that gave students real opportunities to express themselves.

I conducted this study using participant-observation and interviewing over the course of one school year (August 1992 – June 1993). I spent approximately 600 hours observing and working primarily with students in 3 television production classes at Bell High School. During the year I interviewed more than 50 students, many several times. Interviews were both formal and informal. I also interviewed both television production teachers several times each, along with the school principal. I worked with students in several ways — tutoring them on the use of a variety of computer software, helping in the production of credits for videos, acting in one production. In addition, I produced a 15 minute documentary video on the production process which all students viewed, and I developed a CD ROM of student work containing 28 videos, 28 interviews, and 28 commentaries written by the video production teachers.

Data

Data include field notes from observations of 3 video production classes 5 days a week for 16 weeks, more than 125 hours of videotape of students at all stages of the video production process, 90 interviews, written class materials, and more than 100 completed student videos.

Objectives

The ubiquity of print in US schools and society as a whole is such that we rarely consider it as a technology in the same terms as we might view television or computers. Information delivered in print form is everywhere, from street signs to tax forms to newspapers, magazines and books. However, over the past fifty years, print has gradually ceased to be the primary medium of information for many in the US. More and more, video, delivered largely via television, provides news, entertainment and educational material to most of the US population on a daily basis. People in the US spend more time watching television — an average of 58 hours per week — than anything else except sleeping (Schrank, 1986).

While it can be argued that the use of print as a tool for producing texts has not been fully exploited by US schools (Applebee, 1981), the use of video as a composing medium is even less widespread. Few students have opportunities to master the tools of production for video, due in part to the high cost of the technology necessary to create and edit high quality video. In recent years, costs have dropped, and video technology, like print, promises eventually to be incorporated into personal computers as a standard feature over the next few years. At that point, tools for composition of videos or more elaborate multimedia texts including print, video, graphics and animation will be available in the same way that word processing tools are presently available for the construction of print texts.

Research on written composition, while documenting some of the many ways in which print is used by students, has done little to advance our understanding of texts which consist of more than alphabetic symbols. It is one aim of this study to show that theories of written (print) composition provide a starting point for understanding composition with new technologies and new texts, but that these new tools and forms force us at the same time to reconsider the basis of our understanding of reading and writing, and to broaden our view of both process and product with regard to reading and writing.

Theoretical Framework

As print technology has evolved over 500 years, theories of reading and writing have developed to explain our use of print. Over the past twenty years, research on reading and writing has moved from a focus on error and static conceptions of texts and writing skill, to an interest in cognitive models of reading and writing processes, to an emphasis on the social nature of literacy which now permeates current research (Dyson &

Freedman, 1989). What hasn't changed much over this time is a view of reading and writing as separate from the technology of print. To a large extent, reading means reading print and writing means composing print texts. This view has remained fairly constant at a time when the technologies of video and computers give writers access to a wider range of symbol systems and allow them to create a wider range of texts.

Lemke (1989) has challenged the traditional view by positioning reading and writing within the larger domain of social semiotics. From this perspective, reading and writing are both creative, meaning–making activities. While this view may cause us to step back and see print as one part of a larger semiotic system, we need to include as texts those formed out of the wide range of symbolic systems that are available to us, for while Lemke points out the role of intertextuality in meaning–making, he doesn't expand the range of texts that readers and writers are likely to work with. These may include video games, television news, movies, computer–based multimedia environments as well as books and other print forms.

Theories of written composition have begun to explore the wide range of forms and genres used by writers, and to a lesser extent, the range of technologies available to writers. These shifts occur at a time when the computer, having already incorporated and modified the tools for print composition, is about to make tools for video production available to many writers. This convergence of symbol systems will make the range of texts discussed by Witte (1992) even greater, while at the same time changing the process of writing. Video composition currently exists as an area with mature tools evolved over twenty years, and a range of genres. An understanding of video composition techniques will help inform theories of composition, whether they refer to writing, multimedia or video, for ultimately, all of these forms will exist as part of a single environment.

Description: Video Production Process

Videos produced by students at Bell High School may take anywhere from an hour to several weeks or months to complete. As in traditional filmmaking, the process can be seen in three stages: pre–production, shooting, and post–production, but given the social context, the significance and content of these three phases is unique. Although the description that follows is linear, the production process is not. Students may move back and forth between idea and script, writing several versions of a script before showing it to an instructor, or they may use the script merely as a ticket to the production phase and spend most of their time creating the video as they shoot it. Students can also move among shooting, review, and editing as they work. Raw footage may trigger ideas about particular shots that are needed, and editing may point out gaps in a story line that need to be filled in.

Idea

The process begins with an idea and often, a genre (public service announcement, video poem, video essay, music video). The student develops the idea in a one or two paragraph written proposal which describes the aims of the piece and some of the key shots. If a PSA is chosen, the final message is usually included in this proposal. When the proposal is cleared by the instructor, the student is required to write a script describing the proposed video in detail.

Script

A script is a shot by shot description of the images and sounds for the proposed project. Here is an example from a video about AIDS:

Video	Audio
Text on black screen: I don't care about ages...	Original Music
Long shot of young male walking on sidewalk	
Close up of infant in hospital crib	
Close up of elderly female, sitting	

The shot description includes what is on the screen and the kind of shot (close-up, long shot). Often, the camera angle or camera movement (e.g.. side view of people walking.) is included. Despite the apparent level of detail required for a script to pass the script review, there can be a considerable difference between the video as described in the script and the video as shot and edited.

Script Review

The script review is a meeting between instructor and student producer or producers. The written script serves as a "shared space" where teacher and student(s) collaboratively work out the details of the proposed video. This meeting is the major obstacle that must be cleared before students move into the production phase and start working in the visual medium. When a script presents difficulties that cannot be resolved through additional oral description by the student or through acceptance by the student of an interpretation by the instructor of the written text, the script is held for revision. Additional script reviews are often necessary (up to 10 times in some cases) before production may begin.

Shooting

The production phase of the project involves shooting and reviewing video footage. The role of the primary producer here (generally the script writer but in some cases the group leader) may be singular — directing each shot – or involve acting, camera work, and directing all at the same time. The Orson Welles approach (write–act–direct) is actually more common than the Martin Scorcese approach (write–direct), and the scriptwriter who doesn't direct is almost nonexistent. Specialists in acting, camera and post–production work are common though, and work crews who stay together over several productions often develop around these roles.

In some cases, no shooting takes place at all. Instead, students use existing footage to produce their video. Most commonly they search the Video Encyclopedia of the 20th Century for shots by topic, gather the shots and then proceed to edit their video. In other cases, videotapes or live television broadcasts are used as the primary source of footage. Occasionally, footage is borrowed from previous student projects. Students may also combine existing and live footage in their work.

Review

When live footage is being shot, the process usually extends over several days or weeks depending on the length of the script and other factors related to the availability of actors or equipment, the location and time (night/day) of the shots or students' access to areas outside the classroom during the school day. At the end of each shooting period or sometimes during the process and after each shot, students review their footage either with a VCR and monitor or right in the camera. In concert with the review process, students sometimes log their footage to speed up the editing process. In some cases, the review process may end at the close of a class period with the tape advanced only part of the way through the footage, resulting in a loss of footage the next day when shooting resumes and the tape is not checked prior to shooting.

Editing

Editing takes place in a room attached to the main television production classroom. Two editing suites are available, but one consists of an older system that offers no special effects, so it is rarely used at all and almost never used for a final production. (I counted no completed videos done on this system during the 92-93 school year.) The process of learning how to edit is not a formal part of the classroom instruction offered by the teacher. Opportunities to edit arise as students accumulate footage during the production phase, and in some cases the teacher may sit down with a student or students and go through the editing process step by step. This does not happen for every production however, so students more commonly learn how to edit by working with an experienced student and observing them as they work or enlisting their help in completing a project.

Special Effects

Special effects are a part of almost every production at Bell High School. As one instructor put it, students describe videos which do not employ effects as "cheap." Basic effects such as "painting" the video image, changing the color (generally to black and white), or strobing the moving images are available through a special effects board attached to the edit controller. A more extensive library of transitions, effects, and titling is accessible through an Amiga computer equipped with a Video Toaster board. Operation of the Toaster involves the use of a complex software/hardware combination which only a few students have mastered. These experts also tend to be the most skilled editors and often get involved with projects done by other students as helpers at the editing table. In some cases, these students serve primarily as technicians for other students who have a clear vision of what they want. In other cases, the technical assistant may become more involved in the decisions (editing, use of effects, selection of shots) which shape the final video.

Music

Students generally use popular music for the video soundtrack. In many cases, the song is chosen specifically to fit the message of the video. The connection between the song lyrics and the student's message may be very strong or at a more abstract level. In some cases, songs that are narratives may be the basis for the script as well as the message.

During the '92-'93 school year, one student provided original music for several videos. Working with the student producers, he would either compose at a keyboard while watching the video or watch the video several times and create the music at home. The finished soundtrack would then be added to the videotape using editing equipment. In some cases, videos were done initially with a popular song before original music was created. Here, the producers tended to choose music which set a particular mood, then have the student musician create something similar.

Video Premiere

The final event in the video production process is the premiere. When the student has completed the video, it will be shown to each of the 3 video production classes, starting with the class in which the student producer is enrolled. The premiere begins with a brief introduction delivered by the student producer, often detailing the difficulties encountered during the project (actors didn't show up, shots were lost, etc.), describing the personal nature of the video (family member involved in gangs or drugs; personal experience with topic of video), providing a testimonial regarding the extensive help given by other students to help complete the project ("I really have to thank ..."), or pointing out the swiftness of the production ("It took 2 days, really. One day to shoot and 2 hours to edit." Often the length of the production is shortened in this introduction or is described in terms of actual working days rather than calendar days. In the case of many first time producers, no introduction is offered.

The tape is put in the VCR, the lights go out, and the video begins. Darkness tends to prohibit students from engaging in other activities while the video plays, so in general most student eyes are focused on the television screen. In many cases, at the conclusion of the video, spontaneous applause is offered by the class. At other times, if the video does not appear complete no applause is given. In all cases, the instructor begins the discussion with "Say something good about this video." This invitation encapsulates one of the primary goals of the class from the instructor's perspective, to help students succeed and feel good about what they have done. Student praise is generally authentic or not given at all and may focus on general characteristics – "I liked the shots; The music was good; The credits were cool; It was tightly edited," or more specific elements such as the composition of a particular shot – "The shot from the car in the drive-by scene was really cool." Sometimes the praise is offered with a recommendation, as in "That was a good shot but it was a little too long. I think you could have cut it." Once the "good" comments are exhausted, the instructor asks "What could you change about this video to make it better?" Here student response is a little freer than in the praise segment of the reaction where certain elements are generally safe to mention – music, editing, acting. If the student producer is present, negative comments are normally offered with quite a bit of qualification to lessen their impact, or they are not offered at all. The exception here is when the commentor is a good friend of the producer and may offer constructive criticism more directly.

Almost all videos are shown twice, with the same discussion pattern following the second showing with the additional invitation by the instructor, "Did you notice anything that time that you didn't see the first time?" In addition, before the second showing, the instructor may point out specific elements such as camera angle or movement, shot selection, special effects, or editing that he considers particularly noteworthy. Noteworthiness is often related to creativity, so if a new technique has been used for the first time or a different approach to a particular topic is employed, the instructor may point it out.

Following the premiere, most videos are complete. Traditional video editing equipment can be compared to typewriters in the writing process — easy to add on at the end of a text, but very difficult to change anything in the middle. As a result, even if feedback from the video premiere has presented the student with suggestions that might improve the video, these are rarely incorporated into a second edit of the project. In some cases, the audio may be changed, but this process does not require that the video track be modified. When there is the possibility that a video might need reediting, it may be offered to the class as a work in progress, with the stated goal of the premiere to solicit feedback before proceeding with further editing.

The premiere is rarely the first real showing of the video as it has evolved during editing and may have been seen by the instructor at several stages of the editing process. In such cases, individual shots and sequences may have been changed before the video is judged to be complete, but due to the difficulty in performing edits in the middle of completed video, the project generally evolves in a strictly linear process.

Conclusions

The development of a classroom culture that supports and encourages the use of video is a key part of overall success (however measured) of this program. This culture is maintained in part through the unequal distribution of student expertise which supports a small group of students with strong technical and social skills. The group is kept small primarily due to the limited amount of technology available. Unlike computer expertise, many video skills must be developed at school with equipment not available at home. The highly skilled students must develop strong social skills along with technical knowledge (contrast with "computer nerd" culture) in order to learn and complete projects. Team players can do more with video.

This video culture also sustains itself through the creation of artifacts (videos) which are integrated into the curriculum, myths delivered as testimonials by teachers and students, and informal apprenticeships in technology use. Success fosters success as the work of previous classes is used to set high standards for current students. As technology and students have changes, so has the teaching that supports this video culture. Instructional style has evolved over more than 20 years and is marked by willingness to experiment, ability to facilitate rather than instruct, and increasing the intellectual freedom available to students. This style puts both teacher and students in frequent conflict with administration which favors control and status quo in what is perceived as a potentially dangerous environment.

The video production process allows students to develop specializations (acting, technical skills, music) and work collaboratively on projects in ways that are not common in traditional classrooms and the public nature of the video text facilitates authentic assessment via the video premiere, although an informal code limits negative feedback. The premiere serves multiple purposes — marks the completion of a project, serves as inspiration to other students, lets students share their work with an audience of their peers, allows instructor opportunities for teaching during analysis, increases "reading" skills through shared, public analysis of texts.

References

Applebee, A. (1981). Writing in the secondary school, NCTE Research Report No. 21. Urbana: NCTE.

Cuban, L., Teachers and Machines, New York: Teachers College Press, 1986.

Dorr, A. and D. Kunkel, Children and the media environment: Change and constancy amid change, 1990.

Dyson, A. H., & Freedman, S. W. (1991). Critical challenges for research on writing and literacy (Technical Report No. 1-B). Center for the Study of Writing and Literacy.

Koenig, A. E., The development of educational television, Madision: University of Wisconsin Press, 1969.

Lemke, J. L. (1989). Social semiotics: A new model for literacy education. In D. Bloome (Ed.), Classrooms and literacy (pp. 289-309). Norwood, NJ: Ablex.

Mander, J., In the absence of the sacred, San Francisco: Sierra Club, 1991.

Postman, N., Amusing Ourselves to Death, New York: Penguin, 1985.

Schrank, J. (1986). Understanding mass media. Chicago: NTC.

Witte, S. P. (1992). Context, text, intertext: Toward a constructivist semiotic of writing. *Written Communication*, 9(2), 237-308.

Simulated Conversations: The McGill Negotiation Simulator

JOHN ROSTON
Instructional Communications Centre
McGill University, 550 Sherbrooke Street West, Suite 400, Montreal, Quebec, H3A 1B9, Canada
E-mail: johnr@550sherb.lan.mcgill.ca

Abstract: The ability to successfully conduct specialized conversations in stressful situations is an essential part of professional competence in a number of subject areas, but there are few opportunities for students to practice the required skills in a realistic environment. The McGill Negotiation Simulator is an interactive full motion video project with the objective of providing realistic, involving and challenging simulated conversations between the student and someone who appears on the computer screen. The Simulator is described and the degree of realism is discussed. The conceptual and implementation problems encountered in its development are related and there is a brief summary of the advantages and disadvantages of working with consultants.

The Need for Simulated Conversations

The ability to successfully conduct specialized conversations in stressful situations is an essential part of professional competence in a number of subject areas. The purpose of the conversations may involve imparting or eliciting information, establishing a personal relationship, persuading, or a combination of these and other elements. Specialized conversations are central to a wide range of subject areas including, for example, the health and social sciences, management, education, religion and anthropology.

In most of these subject areas students are given formal instruction in conducting specialized conversations. This often includes observing conversations conducted by others either in person or on videotape. Students may also have the opportunity to engage in role-play exercises with other students followed by supervised practice in the field although frequently the former is less than realistic and the latter is difficult to arrange. In both cases, the instructor has very little control over the nature of the experience.

In 1990 the McGill Negotiation Simulator project was established to investigate using interactive video to simulate these types of specialized conversations. The idea was to offer a new and different experience to the student rather then to replace the existing use of role-play exercises and supervised practice in the field. One advantage of a simulated conversation is that it can be repeated a number of times so that the student can experiment with different strategies for handling the same situation. A conversation can be replayed exactly as it took place with feedback to the student on each choice which was made.

To maximize the usefulness of the Simulator, it was decided to create a series of increasingly challenging conversations with the same person on the same subject by adding new facts to the situation under discussion in the first conversation. Insofar as the student has already learned the basic facts of the situation, less time is needed to prepare for the new conversation.

Creating a Realistic Simulation

In a simulated conversation, someone appears on the computer screen and speaks directly to the student as shown in Figure 1. The student replies by choosing what to say from a menu which appears on the screen as shown in Figure 2. Unlike a role-play exercise, the student cannot say anything he or she wishes. On the other

Figure 1. Speaking directly to the student

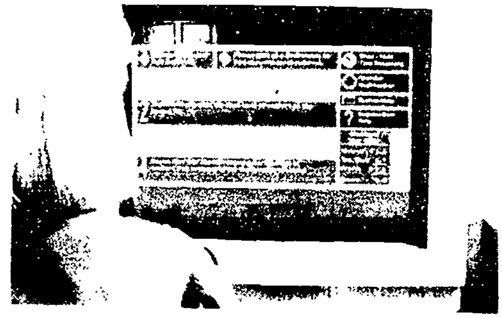


Figure 2. Choosing a reply from the menu

hand, the instructor has complete control over the person who appears on screen.

The degree of realism required is open to debate. It has been suggested that the required level of fidelity in simulations is directly related to the level of expertise of the student (Alessi, 1988; Garhart, 1991). As the student's knowledge level increases, higher fidelity simulations are required.

Maximizing realism may not always maximize the learning experience. For example, we were initially very concerned about the fact that the student chooses what to say from a menu rather than being free to say anything at all. However in practice this seems to force the student to concentrate on the matter at hand and on choosing when and how to use the skills which we are trying to teach. In role-play exercises, it is much easier for students to forget about some of the skills altogether or fill whatever time is available with idle conversation.

In this case, it was decided to maximize fidelity by using high resolution, full screen, full motion video for two reasons. First, the final and most challenging conversation in the series would presumably require the highest level of fidelity. Second, there was considerable doubt that the simulations would turn out to be sufficiently realistic even with the best video quality so there was great reluctance to settle for something less.

Full screen video has the advantage of removing distractions from the screen and focusing the student's attention on the person speaking. Realism is enhanced by always keeping the person on screen. After speaking, he or she waits for an answer rather than disappearing. This is what happens in a real conversation. The other person sits there waiting for a reply.

The obvious disadvantage of high resolution, full screen, full motion video is hardware cost. Although the cost of hardware with this capability is likely to decrease rapidly over the next few years, it currently makes little sense to create expensive simulations in subject areas where potential users cannot afford to use them.

The teaching of negotiation skills is an important component of both graduate management courses in universities and executive training courses in major corporations. In real life, negotiations are often conducted in a high stress environment where the cost of negotiation mistakes can be considerable. Although the hardware for a negotiation simulation may be expensive, it can result in cost effective training.

Hardware and Software

The choice of *TenCore Language Authoring System*, from Computer Teaching Corporation, as the authoring software was made on the basis of recommendations from authors in other university audiovisual centres. Its main advantages are its high degree of flexibility and its extensive debugging features. The latter have proved invaluable in tracking down the author's frequent bumbling in the middle of very complex code. *TenCore* supports a wide range of hardware for the PC platform including both videodisc players and DVI. A lot of time was spent evaluating *DVI* and other forms of compressed video on CD-ROM before concluding that they cannot yet meet our requirements.

In discussing the Simulator with potential users at other institutions, its ability to run unattended was frequently questioned. What if someone puts in the wrong disc? What if there is a power failure? What if a student plays around and manages to delete files? The latter problem is the most serious and was solved by locking away the keyboard. Everything can be controlled by using the mouse.

Solving some of the other problems necessitated learning in detail how video overlay boards work and how videodisc players are controlled. Although *TenCore* has a driver for each overlay board that it supports, it also allows the author to communicate directly with the board. This turned out to be very important since authoring language board drivers generally support only one video mode and a subset of board commands. For example, we wished to have a user code recorded on our videodiscs which could be read by the program to ensure that the student has the right videodisc in the player. It turned out to be surprisingly difficult to get full information on how user codes work and how particular players read them.

Major Problems Encountered

It was decided that the first series of three simulations would involve a sales negotiation in which the student represents an aircraft manufacturer and is trying to sell a commuter aircraft to the Vice-President of a major airline. There were problems on two different levels. On the conceptual level, the question was whether simulating a person was possible given the incredible complexity of human communication. Further, how could we be sure that this person would behave as most executives would in similar circumstances? In addition, what assumptions were we making in deciding on the correct way for the student to respond during the negotiation?

On the implementation level, the question was whether the student would have the feeling of carrying on a conversation with a real person or a stilted exchange with a cartoon character generated by a machine. If the person would remain on screen while the student chose what to say from a menu, how could this be accomplished without using vast quantities of video time?

Conceptual Problems

Most simulations model a process which is well understood and predictable. Whether the process involves the control of equipment or other resources, the authors of the simulation can be confident that they know what would happen for any course of action taken by the student. Moreover, there is a right and a wrong way of proceeding in any given circumstances.

As authors we might have our own ideas of how an average executive would respond in a particular negotiation situation, but human beings are by nature unpredictable. On top of that, negotiation is not an exact science. Although a particular course of action may clearly be a mistake in a given situation, there is likely to be more than one good way to proceed. What might work well in negotiating with one person could fail if used with someone else.

The key to simulating conversations is to realize that you are simulating a particular person on a particular day. He or she might react differently tomorrow. In fact you may wish to have a different version of the same simulation which the student can try tomorrow. Once you realize that there is no such thing as a universally "correct" person, the burden of simulating such a person is removed.

In many subject areas there are multiple approaches or conflicting views on the topic. Students are encouraged to develop their own views by reading and comparing what different authors have to say on the subject. The same thing will occur when negotiation simulations become more commonplace. Students will try our simulations and then compare them to those of other authors.

Insofar as our simulation involved the student choosing what to say from a menu of up to four choices, we decided to frequently have more than one good choice on the menu. Thus there are several different paths to a successful outcome. By allowing students to try the simulation more than once, they can explore different approaches to the same situation whether or not they have done well on the first try.

Implementation Problems

Full motion video is a precious commodity and we could not have long sequences which show the person on screen waiting for the student to choose what to say from a menu. A new technique was therefore devised to create the illusion that the person on screen is sitting there waiting for a reply without actually showing the person. The technique involves seating the executive who appears on screen behind a desk and having him look directly at the camera. After the executive speaks to the student, the menu appears as a horizontal band across the middle of the image. This hides the executive, but leaves visible the desk below the menu and the background of the room above the menu. It creates the illusion that the executive is just behind the menu waiting for the student's response. As soon as the student makes a dialogue choice, the menu disappears and the executive responds appropriately.

A sample videodisc sequence was produced early in the development process and testing on a small number of students quickly revealed that the technique creates a very involving simulation. Further experiments have indicated that several factors greatly enhance the illusion of a real conversation.

It is important that the background image of the room does not disappear or change at any point during the conversation. Otherwise the seamless interaction is broken and the mechanics of the system intervene. We therefore use an overlay card which has a framestore. When the executive finishes speaking, we put up the menu to hide him and then freeze the background using the framestore. When the student makes a dialogue choice, we first find the appropriate response sequence on the videodisc and begin playing it, then we release the framestore as the menu disappears. To avoid a jump in the image of the room at that point, it is important that all of the sequences be shot with exactly the same framing. This requires great care in avoiding any movement of the camera during shooting.

Although the video is frozen while the student makes a dialogue choice from the menu, the audio is not. The audio consists of background office sounds. If the student does nothing for a few minutes, there is the sound of the person behind the menu shuffling his papers and then coughing. This audio is recorded on the videodisc's second audio track. If the student still does nothing, the program switches to a video sequence. The menu is left on screen to hide the person, but his hand reaches forward under the menu and he drums his fingers on the table.

Another important factor in maintaining a seamless interaction is to ensure that the person on screen responds quickly to the student's dialogue choice as he or she would in real life. Quick response time is dependent upon the layout of the videodisc and the search speed of the videodisc player. By using a videodisc player with a maximum search time of two seconds, we can ensure a response time of about one quarter of a second through careful videodisc layout. This is the time required to get the appropriate sequence running and remove the menu. We have discovered that the student then needs time to focus on the image of the executive before being able to concentrate on what he is saying. Each sequence is therefore shot so that the executive pauses for one second before beginning to speak.

Pedagogical Objectives and the Topic of Conversation

The purpose of a simulated conversation is to enable the student to practice specific skills. Although the student must use more than one skill at any particular point in the conversation, we believe that it is important to focus on one skill at a time in order to maximize learning. It is similar to physical exercise where a particular exercise requires the use of a number of muscles, but there is a focus on the muscle which does the bulk of the work.

Insofar as the conversation would cover a number of topics, we decided to associate a particular negotiation skill with each topic of conversation. The dialogue choices available to the student focus on the use of that skill. The instructor can control whether the negotiation skill is communicated to the student at the beginning of each topic and whether there is a help button available to assist the student in evaluating each dialogue choice with reference to that skill. The instructor can also determine whether the student must cover all of the topics in a predetermined order or whether the student can cover the topics in any order and leave topics out. If the student does not choose, or is not allowed, to change the topic, the executive on screen will change the topic as he sees fit and bring up topics which the student would rather not discuss. In real life there are usually topics which are best avoided and the skill associated with discussing them involves extricating oneself as quickly as possible.

We tried a number of ways of allowing the student to change the topic of conversation. Initially we allowed the student to change the topic at any time. This was chaotic and not very realistic. In a real conversation, there are specific points where a change of topic might be appropriate. We have taken those points, and a few others where the student might be looking for a way out of serious trouble, and provided a dialogue choice which leads to a submenu containing the topics which have not yet been discussed. The student can then choose one of these topics and suggest that they move on to discuss it. The executive on screen will usually agree unless the student is in trouble and is trying to change the topic in order to avoid answering a difficult question.

Special Features Required for a Sales Negotiation

One of the topics of discussion is a price negotiation during which the student can change his or her offer and choose what to say in presenting the new offer. The person on screen then replies and may or may not make a counter-offer. The question was how to convey numerical information between the parties. In the case of the student, this turned out to be relatively simple. The student is given a dialogue choice which leads to a new offer work sheet. The student uses up and down buttons to change the price of the aircraft and then confirms that this new offer is to be submitted. New dialogue choices then appear and the student decides on how to present the offer, such as by saying, "This is the lowest I can go."

In the case of a counter-offer by the person on screen, he points to a spot to one side and makes a comment about his counter-offer, such as by saying, "I have raised my counter-offer. Now I want to see a change in your position." We use the computer to generate a card which gives the details of his counter-offer and overlay it in the spot to which he is pointing. It looks as though he is using a flip chart to convey the information.

Enhancing the Realism of the Conversational Situation

Although we have not done any objective testing to determine the relationship between the perceived realism of the simulations and the amount of learning to apply negotiation skills which takes place, we believe that students devote more effort to an exercise which they find to be both challenging and realistic. We believe that the realism of the simulations has been enhanced by using many real elements in what would otherwise be a totally fictitious negotiation situation. As the representative of ATR Aircraft, the student is trying to sell an ATR-42 commuter aircraft to Olympic Airways. Both of these are actual corporations. ATR is a major manufacturer of commuter aircraft which is jointly owned by the governments of France and Italy. Olympic Airways is the national airline of Greece which in fact uses the ATR-42. With the cooperation of the corporations involved and the Government of Greece, dramatic video material was obtained to enhance the presentation of the situation.

The student can consult additional full motion video research material before or during the negotiation session. The display of this material includes control buttons at the bottom of the screen similar to those of a

Figure 3. Doing research on the ATR-42

videocassette player as shown in Figure 3. The student can fast forward or fast rewind. There is also a "Text Mode" button which stops the video and instead displays the narration as printed text on the screen. Although students are told that they can take notes, all of the information which they are given can be repeated by using the appropriate buttons on the menus. In the limited testing so far, students have been about evenly divided in their preference for seeing the video with narration or reading printed text. Most students take very few notes. Student preferences do not appear to be indicative of their success or failure in the negotiation.

Working with Consultants

It would make little sense to devote so much time and money to developing a simulation which could only be used by its principal authors. Early in the development process, a test sequence was shown at a conference on negotiation skills. Its favourable reception led to the recruitment of consultants who teach negotiation skills at seven universities in the United States and Canada. These consultants have been an important factor.

We did not anticipate how much the consultation process would lengthen the development schedule. Flow diagrams which document the simulation are complex and the authors had developed a shorthand method of discussing what should happen at any particular spot. When these flow diagrams were sent to consultants, long written explanations had to be prepared which clarified what each diagram was about and how it fitted into the whole negotiation session.

On the other hand, the usefulness of the simulator improved dramatically as a result of comments made by the consultants. Teaching styles vary widely and the requirements listed by the consultants were sometimes mutually exclusive. A list of parameters was therefore developed which could be set by the instructor in advance to customize the simulation. For example, one parameter controls whether the person on screen will end the meeting and walk out if the student irritates him sufficiently. Although these parameters increases the complexity of the programming, they maximize the flexibility of the Simulator.

Conclusion

It is possible to create realistic, involving and challenging simulated conversations which can be used by students to practice specialized conversational skills. While these have a number of advantages over role-play exercises, they do not replace them. Although the degree of realism required in the simulation is open to debate, every effort has been made to maximize realism by using full screen full motion video and a situation based on actual places and corporations. There has been no attempt to create a universally "accurate" or "correct" person on screen. The student deals with a particular person on a particular day.

A separate conversational skill is associated with each topic of conversation to help the student focus on using that particular skill. Techniques have been developed to enable the student to change the topic of conversation, to allow the parties to exchange financial information and to create the impression that the person on screen is always present waiting for the student's response. Although the use of consultants has lengthened the production schedule, it has resulted in a much more flexible simulation through the development of parameters which can be set by the instructor.

References

Alessi, S.M. (1988). Fidelity in the design of instructional simulations. *Journal of Computer-Based Instruction*, Spring, 40-47.

Garhart, C. (1991). Simulations: how real is real enough? *Journal of Interactive Instruction Development*, 4(2), 15-18.

Acknowledgements

The co-author of the simulation is Professor David Saunders of the Faculty of Management at McGill. Staff of the Instructional Communications Centre have also made many contributions to the quality of the simulation.

SMALLTALKER: A Cognitive Apprenticeship Multimedia Environment for Learning Smalltalk Programming

YAM SAN CHEE
Department of Information Systems and Computer Science
National University of Singapore, Lower Kent Ridge Road, Singapore 0511
E-Mail: cheeys@iscs.nus.sg

Abstract: Cognitive apprenticeship has become increasingly important as an instructional methodology due to a shift by many researchers to the situated cognition paradigm. We describe *SMALLTALKER*, a learning environment that makes extensive use of multimedia to support the learning of Smalltalk programming. The design of *SMALLTALKER* is based on the principle of cognitive apprenticeship. We illustrate how modeling, coaching, scaffolding and fading, articulation, reflection, and exploration have been implemented in *SMALLTALKER*.

In recent years, cognitive apprenticeship (Brown, Collins, & Duguid, 1988; Burger & DeSoi, 1992; Collins, Brown, & Newman, 1989) has become increasingly prominent as a model of instruction. This development is attributable to its potential to help solve the educational problems of brittle skills and inert knowledge that so often arise with traditional schooling (Resnick, 1987; Whitehead, 1929). Recent research in the learning sciences coupled with a shift to the situated cognition paradigm in the cognitive sciences has led to a significant rethinking of the nature of learning and cognition (Brown, et al., 1988; Clancey & Roschelle, 1991) and how we can use technology to support learning (Brown, 1985; Clancey, 1992; Collins, 1991). Cognitive apprenticeship is one manifestation of this new way of thinking about learning.

In the next section of this paper, we describe cognitive apprenticeship as an instructional methodology and explicate its underlying rationale. The following section describes *SMALLTALKER*, a learning environment for Smalltalk programming, and outlines the way in which cognitive apprenticeship has been embodied in the system. We conclude by commenting on the effectiveness of the system in field testing to date and charting the direction in which *SMALLTALKER* will continue to evolve.

Cognitive apprenticeship

The cognitive apprenticeship instructional methodology, as formulated by Collins, Brown, & Newman (1989), consists of six teaching methods: *modeling*, *coaching*, *scaffolding*, *articulation*, *reflection*, and *exploration*. The six methods, in turn, break down into three groups. The first group—modeling, coaching, and scaffolding—is designed to help students acquire an integrated set of cognitive and metacognitive skills through observation and supported practice. The second group—articulation and reflection—is designed to help students gain control of their own problem solving strategies. The final group—exploration—is intended to encourage learner autonomy and problem formulation by the self.

Cognitive apprenticeship embeds the learning of knowledge and skills in their social and functional context. In *modeling*, an expert performs a task so that students can observe and build a conceptual model of the processes required for task accomplishment. The provision of a conceptual model contributes significantly to success in teaching complex skills without resorting to lengthy practice of isolated subskills. In cognitive domains, unlike traditional apprenticeship settings, modeling often necessitates the externalization of internal cognitive processes. Tacit processes are brought into the open so that students can observe, enact, and practise the requisite skills.

In *coaching*, students are engaged in problem-solving activities that require them to appropriately apply and actively integrate subskills and conceptual knowledge. In this way, conceptual knowledge is exemplified and situated in the contexts of use, thereby grounding the knowledge in experience and making learning meaningful. Consequently, this approach helps to avoid learning outcomes where knowledge remains bound to surface features of problems as they appear in textbooks and is incapable of transfer. The expert coaches students by providing hints, feedback, and reminders, thus assisting them to perform closer to his standard of skill. Coaching requires highly interactive and situated feedback. Hence, the content of coaching interaction is related to specific problems that students face in carrying out a task.

In *scaffolding*, an expert assists students to manage complex task performance. If necessary, he completes those parts of the task that students have not yet mastered. This method may entail students engaging in legitimate peripheral participation (Lave & Wenger, 1991); that is, students participate in the practice of an expert, but only to the degree that they can handle and with the amount of responsibility that they are capable of assuming. Scaffolding is coupled with *fading*, the gradual removal of the expert's support as students learn to manage more of the task on their own. The interplay between observation, scaffolding, and increasingly independent practice aids students in developing the metacognitive skills of self-monitoring and self-correction and in achieving integrated skills and knowledge characteristic of expertise.

In *articulation*, an expert encourages students to explicate their knowledge, reasoning, and problem solving strategies. Such activities provide the impetus for students to engage in the refinement and reorganization of knowledge. The use of synthetic and design tasks in a producer–critic framework is particularly effective in achieving articulation (Allen, 1992; Pea, 1991). Such tasks require students to participate in generating knowledge and evaluating the outcomes of knowledge-building as part of collaborative learning activities. Generative and evaluative processes provide a further basis for concept assimilation and internalization.

In *reflection*, the expert provokes students to compare their problem solving processes with his own, with that of other students, and with an internal cognitive model of the relevant expertise. Such comparisons aid students in diagnosing their difficulties and in incrementally adjusting their performance until they achieve competence. Reflection is facilitated by the provision of abstracted replay that contrasts students' own performance with that of the expert (Collins & Brown, 1988).

In *exploration*, the expert pushes students to be independent learners. Students are only set general goals. At the same time, they are encouraged to identify personal interests and pursue personal goals. Forcing students to engage in exploration teaches them how to frame interesting questions and to identify difficult problems on their own.

The *SMALLTALKER* learning environment

Overview

SMALLTALKER is a Macintosh-based multimedia learning environment for Smalltalk programming. The learning environment supports both concept learning and skill acquisition. A special effort has been made to apply the teaching methods of cognitive apprenticeship in designing and implementing *SMALLTALKER*. When the system is launched, an animation of floating Smalltalk balloons is played repeatedly, together with music, until the mouse is moved. After obtaining basic information about student users, the system leads them through an orientation of the system's interface to allow them to begin interacting effectively with the system. The use of various elements of the interface (eg. mouse, windows, and text manipulation) is demonstrated to students via movie clips. Students are then coached on the use of the interface in a different problem setting.

Having obtained basic interface skills, students are introduced to the important Smalltalk programming concept of message-passing. At this stage, students are taught to view object-oriented programming entirely in terms of objects communicating with one another by message passing. This view is an external view (LaLonde & Pugh, 1990) where programming is treated entirely in terms of message-passing syntax (Shafer & Ritz, 1991). Details of method implementation are hidden. The system's interaction with students is structured so as to reify the relationship between process and product. Thus, representations of objects that students interact with are shown graphically, and message passing invokes changes to objects that are visually representable.

After message passing has been covered, instance methods are introduced. This section opens up the internal view (LaLonde & Pugh, 1990) and its concomitant method-definition syntax (Shafer & Ritz, 1991). The instruction then progresses broadly through the topics of instance variables, classes, class methods, class variables, and superclasses. New classes and new concepts are introduced as and when they are needed in relation to the current goal. In this way, progression through a rigid sequence of topics is avoided. We are currently working on the topic of developing a small application. We plan then to treat the Model–View–Controller paradigm in the context of constructing an authentic application.

In the subsections that follow, we describe how each of the teaching methods has been instantiated in the learning environment.

Modeling

SMALLTALKER is unique in that it fulfills *two* roles: that of instruction presenter and that of coach. In its capacity as instruction presenter, *SMALLTALKER* makes extensive use of modeling. To support enculturation, QuickTime™ movies are used to show an expert presenting information to students. MacroMind Director™ animations are used to present instruction on concepts and skills necessary for Smalltalk programming. The movies are accompanied by the expert's voice narration of the instruction.

Figure 1 provides an illustration of how modeling occurs. The figure comprises four snapshots from a Director animation. In this example, the student is learning about the copy-and-paste technique in text editing. The snapshots show how the expert selects a portion of text (Figure 1a), selects "copy" from the pop-up menu (Figure 1b) resulting in the copied text appearing on a notional clipboard, then selecting "paste" (Figure 1c), thus yielding the pasted text at the insertion point of the selected window (Figure 1d).

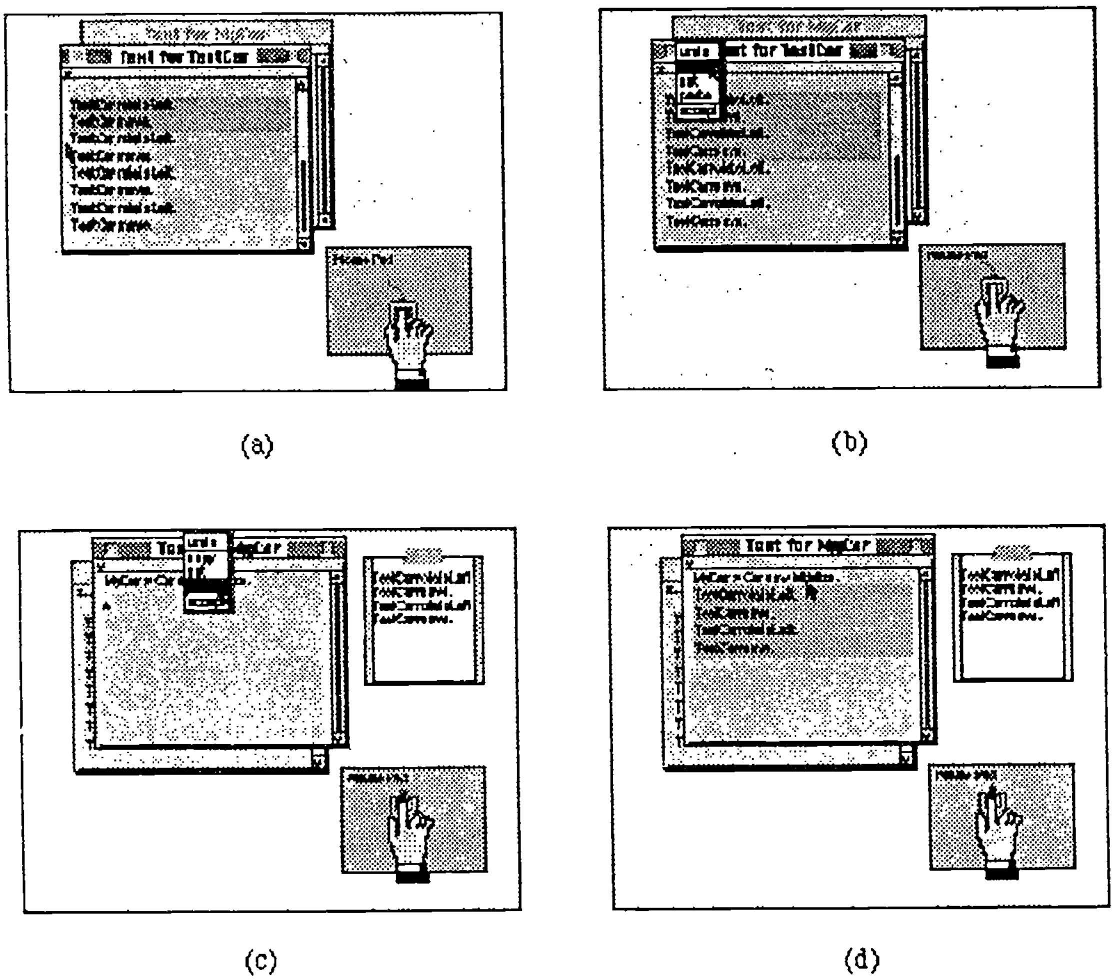


Figure 1 Modeling of copy-and-paste in text editing

Modeling is ideally suited to reifying processes through animation techniques. With the use of multimedia, concepts and processes can be "brought to life," focusing students' attention on the *meaning* and *sense* of what they observe.

In our implementation of modeling, we have framed the learning situation so that students will learn cause-and-effect relationships. Thus, the goals related to a sequence of actions are made explicit *before* the actions are shown, and the results of the actions are shown in a concrete fashion. Our instruction has also been designed to start with concrete and visual objects (eg. TestCar) before moving to more abstract and functional objects (eg. MyRecord) to aid students in grasping the necessary concepts.

Coaching

Coaching is a critical activity in any learning environment because it is here that the teacher or system engages in actions to facilitate students' mastery of problem solving. In *SMALLTALKER*, students are made to work on programming problems while they are still being introduced to new concepts and procedures. The most pervasive method of coaching takes the form of feedback to student actions and errors while they are engaged on programming problems. As an example, a dialog box like that shown in Figure 2 will appear if a message expression coded by a student does not begin with the name of an object. Notice that the feedback given is highly situated; it is specific to the particular programming problem being solved.

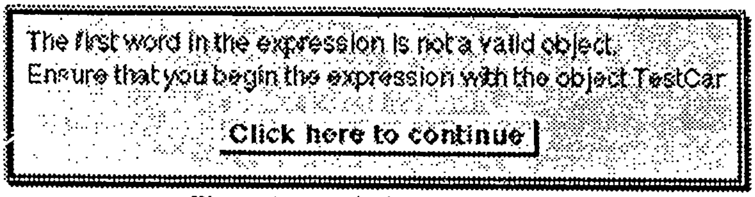


Figure 2 A typical feedback dialog

Besides the use of dialogs, feedback is also given by providing visual animation of objects where appropriate. As illustrated in Figure 3, the effect of executing the code written by a student is made self-evident from the animation of the car moving about in the TestCar window. Thus, students are encouraged to form and test hypotheses, diagnose their own performance, and adopt an active attitude to learning. In this way, students are encouraged to develop metacognitive skills.

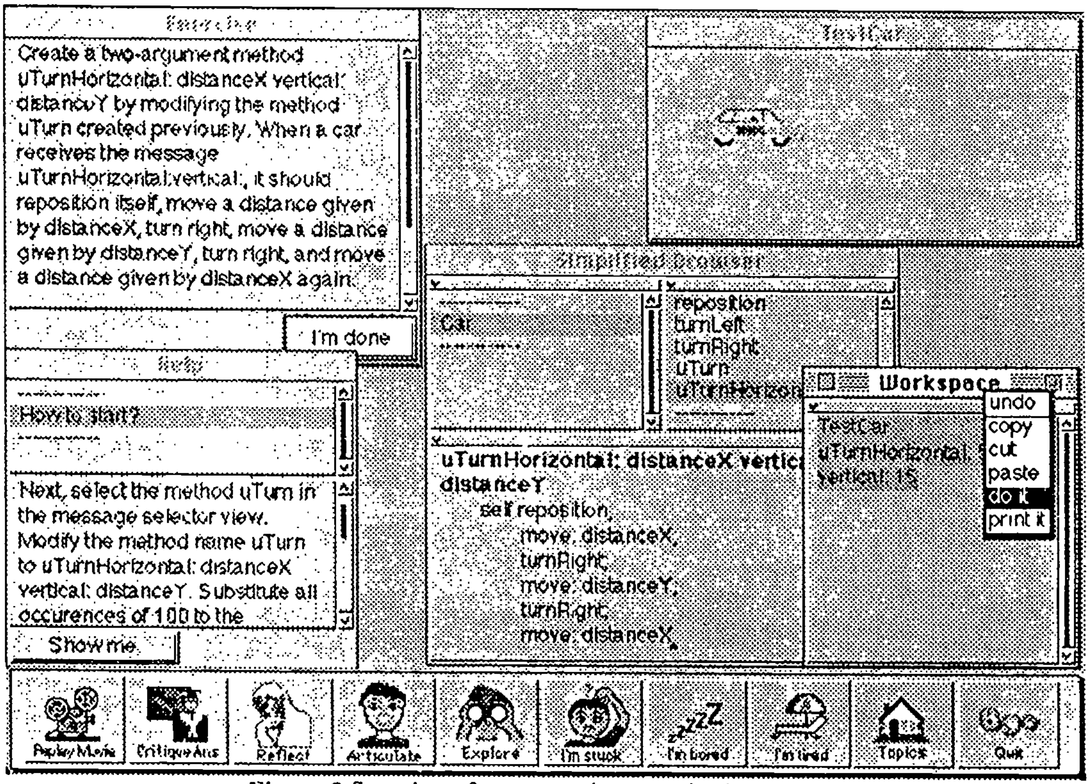


Figure 3 Snapshot of programming exercise in progress

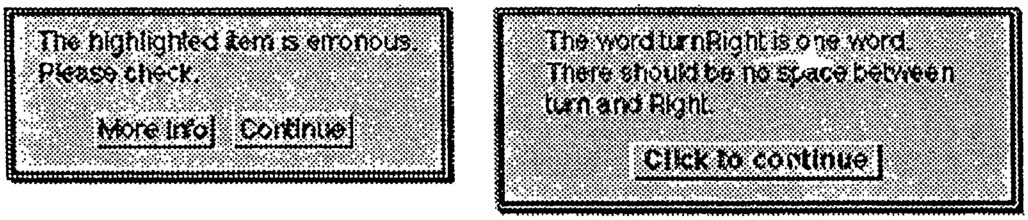
It is important in cognitive apprenticeship that the efforts of students be useful and used. To this end, problem solving exercises often ask students to make enhancements or to add new methods to existing code so that the resulting whole possesses greater value after such changes.

Scaffolding and fading

Scaffolding and fading are difficult teaching methods to implement because they require a teacher to be sensitive to the detailed needs and difficulties of students engaged in task performance. Our approach to this difficulty has been to place the burden of responsibility on the shoulders of students. *SMALLTALKER* provides a fairly open-ended help system (see Figure 3). Students have the option, while problem solving, to request help by selecting the button "I'm stuck" from the panel of buttons at the bottom of the screen. The help window lists a set of topics on which assistance on specific aspects of the problem students are attempting is available. Making the activation of the help system the responsibility of students encourages them to take active control of their learning. A "Replay Movie" button allows students to review the instructional content related to a problem, whenever they wish.

The "Show me" button in the Help window (see Figure 3) is particularly significant. It functions as a student's last recourse. For this reason, it is separated from the help topic list. We have implemented the "Show me" button such that it only becomes available after the student has selected some percentage of the help items available. This approach prevents students from choosing the easy way out of a problem solving situation. When students eventually select the "Show me" button, a movie plays, showing how an expert would have solved the programming problem. This action takes us back to the modeling method.

Fading has been implemented in the kind of error trapping that the system engages in. During the course of early problem solving, special checks are incorporated to prevent students from committing errors from which they would be unable to recover. However, as students become more familiar with Smalltalk programming, feedback dialogs are generalized so as to provide less problem solving support for them. Figure 4a illustrates a feedback dialog with fading while Figure 4b illustrates a dialog without fading. However, even with fading operational, students have the option of requesting a more specific dialog if they feel that the level of support received is inadequate in any particular situation. All that they have to do is click on the button "More Info".



(a) (b)

Figure 4 Fading in feedback dialogs. (a) More general. (b) More specific.

Articulation

Computers are ill-suited to handling the natural language articulations of humans. Nevertheless, we attempt to encourage students to articulate their knowledge by posing questions that request students to articulate answers to various conceptual questions, either to themselves or to a friend. When a student clicks the button "Articulate", an "Articulation" window appears (see Figure 5). Clicking on the button "Tell me more" plays a QuickTime movie explaining the purpose of articulation. Clicking "Proceed" takes the student to a specific question the answer to which requires articulation.

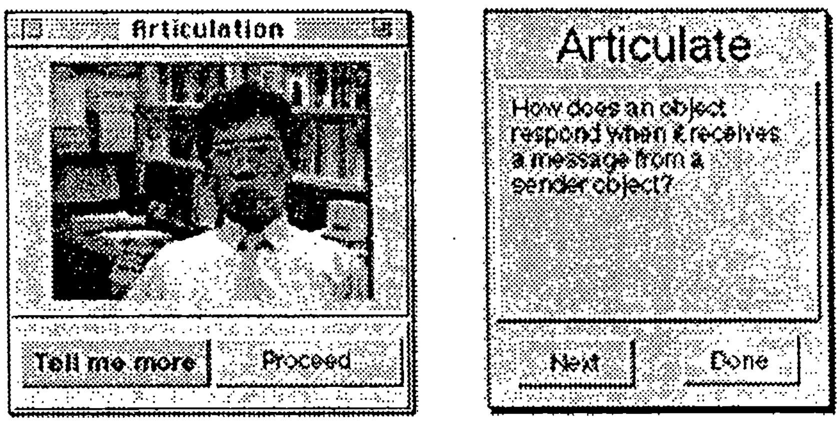


Figure 5 Illustration of the teaching method articulation

Reflection

We attempt to provoke reflection by periodically posing questions that possess deeper conceptual significance. Some reflection questions are included as part of the flow of instruction and, hence, are compulsory. Other questions are optional and can be called up by clicking on the button marked "Reflection". After students have reflected on the question posed, they can click on the button "Play" to listen to an expert expressing his view on the question (see Figure 6). In this way, students can gauge to what extent they have come to appreciate the subject domain in the way that an expert does.

Exploration

It turns out that exploration is an activity that system designers cannot prevent students from engaging in if the system succeeds in gripping their interest and arousing their motivation to learn. We support students'

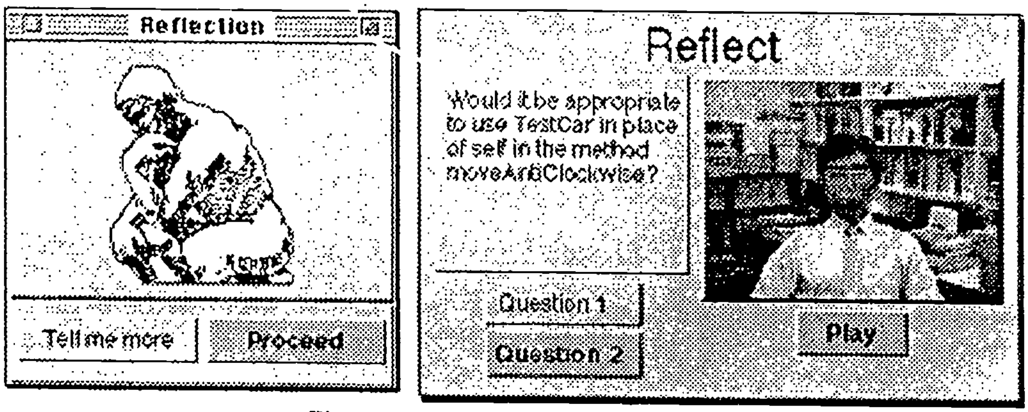


Figure 6 Reflection in *SMALLTALKER*

exploration of Smalltalk programming by providing them with an "Explore" button. Clicking on this button takes students into the ParcPlace Smalltalk–80 programming environment. A "Return to *SMALLTALKER*" button allows students to return to *SMALLTALKER*.

Conclusion

Our evaluation of the *SMALLTALKER* implementation to date has been encouraging. In general, students have found the learning environment extremely user-friendly and very supportive of their learning needs. *SMALLTALKER* only attempts to provide students with a first course in Smalltalk programming. Given the complexity of Smalltalk programming and the need to support the development of programming skills as part of a broader community of practice, we have embarked on the development of a complementary learning environment *CALET*, the *C*ognitive *A*pprenticeship *L*earning *E*nvironmen*T*, that has as its objective the provision of situated Smalltalk programming cases for realistic "on-the-job" programming practice.

References

Allen, C. L. (1992). Multimedia learning environments designed with organizing principles from non-school settings. In E. De Corte, M. C. Linn, H. Mandl, & L. Verschaffel (eds.), *Computer-Based Learning Environments and Problem Solving* (pp. 465–484). Berlin: Springer-Verlag.

Brown, J. S. (1985). Idea amplifiers: New kinds of electronic learning environments. *Educational Horizons, 63*, 108-112.

Brown, J. S., Collins, A., & Duguid, P. (1988). *Situated Cognition and the Culture of Learning*. Tech. Rep. No. IRL 88-0008. Institute for Research on Learning.

Burger, M. L., & DeSoi, J. F. (1992). The cognitive apprenticeship analogue: a strategy for using ITS technology for the delivery of instruction and as a research tool for the study of teaching and learning. *International Journal of Man-Machine Studies, 36*, 775-795.

Clancey, W. J. (1992). New perspectives on cognition and instructional technology. In E. Costa (ed.), *New Directions for Intelligent Tutoring Systems* (pp. 3–14). Berlin: Springer-Verlag.

Clancey, W. J., & Roschelle, J. (1991). Situated cognition: How representations are created and given meaning. *Educational Psychologist*, forthcoming.

Collins, A. (1991). Cognitive apprenticeship and instructional technology. In L. Idol & B. F. Jones (eds.), *Educational Values and Cognitive Instruction: Implications for Reform* (pp. 121–138). Hillsdale, NJ: Lawrence Erlbaum.

Collins, A., & Brown, J. S. (1988). The computer as a tool for learning through reflection. In H. Mandl & A. Lesgold (eds.), *Learning Issues for Intelligent Tutoring Systems* (pp. 1-18). Berlin: Springer-Verlag.

Collins, A., Brown, J. S., & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L. B. Resnick (ed.), *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser* (pp. 453-494). Hillsdale, NJ: Lawrence Erlbaum.

LaLonde, W. R., & Pugh, J. R. (1990). *Inside Smalltalk, Volume 1*. Englewood Cliffs, NJ: Prentice-Hall.

Lave, J., & Wenger, E. (1991). *Situated Learning: Legitimate Peripheral Participation*. NY: Cambridge University Press.

Pea, R. D. (1991). Learning through multimedia. *IEEE Computer Graphics and Applications, 11*(4), 58–66.

Resnick, L. B. (1987). Learning in school and out. *Educational Researcher, 16*(9), 13-20.

Shafer, D., & Ritz, D. A. (1991). *Practical Smalltalk: Using Smalltalk/V*. NY: Springer-Verlag.

Whitehead, A. N. (1929). *The Aims of Education*. New York: Macmillan.

Using a Layered Multimedia Model to Build Educational Applications *

GERHARD A. SCHLOSS MICHAEL J. WYNBLATT
Computer Science Department
SUNY at Stony Brook, Stony Brook, NY 11794
email: {schloss, wynblatt}@cs.sunysb.edu

Abstract
Currently, educational multimedia (MM) products are prepared in an ad hoc fashion, i.e. the different media objects, when needed, are produced by the MM author. In the future, as more digitized MM data becomes available, building multimedia compositions will likely involve searching through databases of existing images, sounds, and video clips. Where today a MM application is typically associated with a specific hardware platform, future MM developers will author MM products which are reusable across a variety of platforms. A recently introduced layered formalism, which describes MM applications in terms of the data that they employ, provides a structure that interfaces MM data with MM authoring. Such a formalism together with a database of MM data objects, is a convenient development environment for creation of a variety of reusable and portable educational multimedia.

1 Introduction and Background

Two problems faced by the majority of today's multimedia (MM) users point to the fact that future MM application development is likely to become a data-driven process, relying heavily on shared MM data repositories as the main resource for MM authoring. First, multimedia data is storage intensive. Second, the production of high quality audio, video and animation is an expensive and time consuming process, which few potential MM authors will have the skills or resources to master. Hence, the concept of several users sharing access to large MM databases will alleviate both of these problems by distributing storage, and production or purchase costs among several individuals or institutions.

Of all the new digital technologies, multimedia computing seems to be the one with the greatest potential to create interest and attract new users from Social Sciences, the Arts and Humanities. Their involvement and contributions can significantly increase the penetration of computing into the traditionally technology-averse parts our society. However, fewer people in the non-technical community are skilled in programming, and their most important interaction with multimedia computing will likely occur at the interface level [Ward, P. S. and Arshad, F. N., 1991]. Hence the importance of designing a convenient and easy-to-understand logical structure, one that facilitates data transfer from the MM data repository to MM applications.

Recently, we proposed and described a new modeling approach to MM data management and MM application development that follows the *layered architecture* paradigm [Schloss, G. A. and Wynblatt, M. J., 1993]. There are two major reasons that prompted the development of the LMDM model. First, many characteristics of MM data make it uniquely different from the traditional alpha-numeric data. In particular, MM data may have semantics that are only relevant with regard to its presentation. For instance, a clearly sarcastic remark in a sound bite, may not convey its true meaning in text representation.

Second, the existing data models generally do not support *multimedia development* because their prime focus is the data itself. In traditional data models, presentation is usually irrelevant. Indeed, a record with a person's address may be displayed in a large box or a small one, in the Helvetica font or the Times; the meaning of the address is still the same. Compare that with a color photograph which conveys some message that would be obscured by a monochrome display. Hence, our objectives in developing the LMDM were that

*This work was partially supported by the National Science Foundation under Grant No. CDA-9214942.

the new model: (a) be application-independent; (b) allow conceptual separation between the data itself its presentation; (c) provide an interface to a shared MM database; and (d) support a wide variety of MM applications.

At a time when multiple-platform computing environments are becoming the rule rather than the exception, MM applications are among the least portable [Woelk, D. et al., 1986]. This is due in part to their dependence on specific hardware for presentation. LMDM's layered architecture allows description of complex data structures within applications, independent of the particular hardware specifications.

We provide below a concise description of the LMDM, and then discuss its potential as a toolkit for developing flexible, adaptable, reusable and portable educational materials, which utilize interactive multimedia compositions.

2 The Layered Multimedia Data Model (LMDM)

LMDM follows the layered modeling paradigm. Such an approach, in addition clearly separating data management, presentation management, and application development, has several other advantages. First, hardware- and OS-specific concerns can be isolated in particular layers, resulting in a more general model. Second, the implementation of the functionality of different layers can progress independently, so that the advances in one area can quickly be added to existing strengths in others. Finally, a layered paradigm is a clear favorite with those who perceive multimedia as a new communication and information delivery system. A layered MM model was previously proposed in [Klas, W. et al., 1990], though the choice of layers was somewhat different.

The LMDM model consists of four layers, see Figure 1. It places particular emphasis on expressiveness for continuous media data objects like video, animation, and audio, allowing them to be sequenced, synchronized or related temporally in other ways. We provide below brief definitions of the four layers, their functionality, as well as of the basic terminology used in the LMDM.

2.1 Data Definition Layer

The Data Definition Layer (DDL) allows data specification of a *MM data object*, either explicitly or through some reference to the data, and it provides this abstraction to the higher layers. A data object consists of a definition statement which specifies the data itself, and also a data type which describes properties of the data. The DDL provides: (1) a language in which the definition statement may be made formally; (2) a set of types which can be used to describe the retrieved data; (3) a tagging mechanism, through which tags containing semantic information may be attached to data objects to improve data accessibility; and (4) an accounting or charge-back mechanism, through which MM data objects containing information subject to copyright or royalty agreements may be accessed.

Data types convey semantic information. In addition to data types such as audio clip or text passage, more specific data types like `music` or `movie` can be defined. In order to facilitate description of such data types, the DDL supports subtyping and attribute inheritance. New media can be introduced by creating a new data type to represent the media and providing operators to act on objects of the new data type. The DDL also provides an *object scripting* mechanism which allows filtering, enhancing, altering, and comparing data objects, as well as grouping data objects into arrays, without modifying the data itself.

2.2 Data Manipulation Layer

The Data Manipulation Layer (DML) allows data to be combined into more complex constructs called *MM events*. MM events are MM objects, or groups of MM objects, which share the same abstract event clock, or event time reference. The DML provides operators and an algebra in which these transformations can be described formally. It contains a symbolic language for formal description of complex manipulations. Hence, simple MM events can be combined to form complex temporal structures.

Data objects that share a time reference can be related to each other through temporal relationships [Hoepner, P., 1992], which can be used for sequencing (before, after) or synchronization (starts, finishes). Synchronization points may be described, to allow arbitrary strictness of synchronization. Objects which share a time reference are said to be *temporally bound* to each other.

Layer		Type
Control Layer	⟹	MM Composition
Data Presentation Layer	⟹	MM Presentation
Data Manipulation Layer	⟹	MM Event
Data Definition Layer	⟹	MM Object

Figure 1: The Layered Multimedia Data Model (LMDM).

2.3 Data Presentation Layer

The Data Presentation Layer (DPL) provides a description of how data is to be communicated to the user. The DPL uses MM events as its building blocks, adding information about layout, output format, presentation dependencies between events, user interface elements such as windows or icons, as well as default values for playback parameters such as colormap and speaker volume. The resulting *MM presentation* is a set of instructions for communicating MM data to a viewer.

The DPL maintains a library of specifications for the various playback devices, spatial layouts of the data on the screen, windows or other display methods, colormaps, text fonts and styles, as well as descriptions of presentation dependencies between frequently used MM event templates. The DPL also includes the logic necessary to describe the interaction of simultaneously occurring events, e.g. opacity and color of object overlays (alpha channel), and volume adjustment of multiple sound tracks.

In the same way that a MM event may have multiple presentations associated with it, a single MM presentation may contain several different events, or may be reused with a different set of events. This allows data to be presented in an alternative manner if certain hardware or presentation parameters change. Moreover, generic presentations may be developed which can present several similarly structured events in the same way.

2.4 Control Layer

A *MM composition* is defined as one or more MM presentations which are grouped with a common control structure and user-interface. Hence, the Control Layer (CL) describes how such compositions are built from one or more presentations. It contains ordering instructions for the various MM presentations, a navigation mechanism through which the user interacts with each presentation, and conditions for starting and stopping the different presentations that make up a MM composition. The CL determines whether these presentations are logically affiliated, and how they interact with each other and the user.

The CL provides a language which describes what signals the MM composition can accept from the user or from I/O devices, and what actions must be taken upon receiving these signals. These actions might include stopping or starting a presentation, or changing the presentation mode in one or more of the presentations. In addition to describing the reaction to signals, the CL language provides constructs for 'hard-coded' sequencing and looping of presentations, as well as for conditional presentation based upon the values of any available input.

3 LMDM and Interactive Learning

In [Pea, R. D., 1991], a clear distinction is made between the *chained multimedia* [sic] and *integrated multimedia* in education, where the former is attributed to the traditional audio-visual aids, e.g. interruptible slide shows or educational films. We feel that a more computer-oriented terminology may be in order. Hence, we distinguish between: (a) *computer-controlled multimedia* (CCMM); and (b) *computer-integrated multimedia* (CIMM). With CCMM, a MM presentation or experiment may be centrally controlled by a computer; however, the participating media need not be digitized. An example of a system that operates in the CCMM mode is a MM classroom [Skill Dynamics, 1993], where a mix of analog and digital devices can be interconnected and slaved to a computer. Thus, in CCMM the range of possibilities in both data integration as well as user interaction, are strictly limited. In contrast, CIMM implies fully-digitized media, with a complete integration of the computer in *all* aspects of media production, orchestration, storage, retrieval, manipulation, and delivery to the end-user. Hence, the CIMM opens unlimited possibilities to both the

MM materials developer and MM consumer, as any imaginable mix of data, as well as non-linear access and viewing, can now be supported. Since large-scale CIMM systems are not yet technologically feasible, a *MM workstation* equipped with media-editing and MM authoring software is the current surrogate for the future CIMM system.

The specifications that led to the development of the LMDM model, are clearly intended as a framework that would efficiently describe and support computer-integrated multimedia. As many MM compositions have interactive components, the LMDM supports the capability to express the effects of user-interaction. Educational MM applications often require support for sophisticated hypermedia-style browsing and navigation techniques [Ward, P. S. and Arshad, F. N., 1991], which are necessary for self-paced access to learning materials. In LMDM, the users can mark their current position within the application for future return (a technique similar to a *bookmark*), or may designate a collection of bookmarks within the application as a walk-through path that can be stored and repeated at will. This second technique is often called a *trail* [Lipton, R., 1992]. Finally, the support for persistent storage provided within the LMDM makes tools for instructional chores such as testing, quizzing, and score keeping, easy to create and maintain.

The layered structure of the LMDM allows the developer to pick and choose, by tailoring her own development system through personalized selection of tools for each of the LMDM layers. Moreover, the extensive browsing and other score-keeping features, which can be supported with relatively little effort within the LMDM, greatly expand the number of different teaching styles and teaching strategies that can be implemented utilizing LMDM. Likewise, the amount of feedback given to students during instruction is completely flexible, and it can be adjusted as a function of their progress or achievements. Thus, LMDM avoids some of the weaknesses observed in the older computer-based training (CBT) systems, such as HITS [Barker, J. and Tucker, R. N., 1990].

The LMDM allows fast prototyping of many different instructional paradigms. In the following example, we demonstrate how a reusable structure of a tutorial can be built with the help of LMDM. Space limitations prevent us from making the example too complicated.

4 Example: Building a Tutorial with LMDM

Suppose that the objective is to develop a tutorial designed to teach the difference between two things. It might be used to teach the difference between kinds of architecture, or styles of painting, or different makes of car, or any similar distinction. The method of instruction relies on a slide show paradigm accompanied by appropriate narration and text. The student is to acquire the knowledge provided by the tutorial, and be tested before she is allowed to proceed to the next subject. Hypothetical specification examples are provided for each layer.

Data Definition Layer:

At the DDL, we define a MM object called a *lesson*. A lesson is an aggregate [Woelk, D. et al., 1986] and conceptual [Klas, W. et al., 1990] object, which links all the pieces of data that are needed as part of a particular tutoring session. Lesson objects may be custom designed for particular tutorials, but their real power can be shown through *lesson templates*, which allow tutorials covering similar materials to be developed rapidly. A *template* of an object is defined as the meta-data that describes the object structure without explicitly stating the data (contents). When initialized (or, linked) with specific data, a template becomes an instance of the object.

Consider the following object template, called a `SlideShowLesson`. This template exists at the data definition layer, and describes the *types* of data involved in the lesson, without actually describing the data. In this particular example, the `SlideShowLesson` consists of two sets of photographs, and a series of narrations describing the photographs. It also contains information about the preferred colormap and resolution of the photos and the recording rate of the audio narrations. Finally, it contains a link to a *score* object, in which the results of the lesson will be kept. In this example, the score object is defined as generic text.

```
Template SlideShowLesson:
     contains
          photo1: array [1..M] of photos
          photo2: array [1..M] of photos
```

```
            narration: array [1..N] of audioclips
            score: text passage;
      PreferredCmap is worse(photo1.preferredCmap, photo2.preferredCmap);
      PreferredRes is worse(photo1.preferredRes, photo2.preferredRes);
      PreferredRate is narration.RecordingRate
```

Data Manipulation Layer:

Based on this template defined in the DDL, a full composition can be built in the higher layers, without requiring precise knowledge of what the actual content of the photos and narrations will be. At the manipulation layer, photos and narrations have to be synchronized and sequenced.

In this example, narrations are synchronized with the first N pairs of photos. These narrated photos provide the explanatory section of the composition, in which the narrator explains a difference in style by referring to pairs of photos which exhibit that difference. After the explanatory section, there is series of unnarrated photos which are used as a quiz section. The manipulation layer also describes an operator Scoring, which accepts a string and updates the data in the score object.

```
      Event SlideShow is
            Sequence((for I from 1 to N
                        TempEqual(narration[I],photo1[I],photo2[I]))
                  for I from N+1 to M
                        either UntilRestart(photo1[I]) or
                              UntilRestart(photo2[I])
                  )
      Operator Scoring(answer:String, score:TextPassage)
            /* ... definition ... */
```

Data Presentation Layer:

The DPL describes the window in which the presentation will take place, and mentions that the preferred colormaps and resolutions will be used. The presentation has two modes, one for the explanatory section, which has a Next button, and one for the quiz section, which has Style1 and Style2 buttons. The functions of these buttons are specified at the control layer.

```
      Presentation #1
            is SlideShow
            using  PlayAudio(narration)
                        with SlideShowLesson.PreferredRate
            and ShowImage(photo1) and ShowImage(photo2)
                        in 10"x10" SplitWindow
                        with SlideShowLesson.PreferredCmap
                              SlideShowLesson.PreferredRes
            Mode #1 has PushButton "Next" at (150,30);
            Mode #2 has PushButton "Style1" called photo1.stylename
                                          at (150,10)
                  and PushButton "Style2" called photo2.stylename
                                          at (150,50);
```

Control Layer:

The control layer describes the control sequence. The presentation moves forward for the duration of an audio sequence, and then pauses until the Next button is pressed. After the first N slides, the mode is changed from Mode #1 to Mode #2. Now the student advances the presentation by selecting one of the two Style buttons to indicate which of the two styles she believes this photo represents. Her choice is recorded in the score object using the Scoring function provided at the manipulation layer. Note that no bookmarks or trails need to be defined, because in LMDM these services are globally available.

```
      Signals
            Signal Next from Button "Next";
            Signal Style1 from Button "Style1";
            Signal Style2 from Button "Style2";
            Signal AudioEnd from Method PlayAudio;
```

```
Control Script
        for I from 1 to N
                DO Presentation#1(Mode#1) UNTIL AudioEnd;
                WAIT UNTIL Next;
        for I from N+1 to M
                DO Presentation#1(Mode#2) UNTIL Style1 or Style2;
                Scoring(LastSignal,score);
```

In this example, we have designed a complete interactive multimedia tutorial, without actually specifying the subject of the lessons. In order to use the SlideShowLesson, an instructor need simply provide two sets of photos and record a few narrations. The work of putting together the synchronization and control, as well as designing the screen layout and user-interface, is a one time cost, and can be re-used with many different tutorials.

Another advantage of this example is that the score data is persistent, and can be linked to the material of the lesson (the photos) at the definition layer. Not only can the current tutorial use the score to determine what areas need review, but other, independent, tutorials can draw upon this information as well. Other tutoring programs can check to see what material the student has mastered, and what requires more study. In this way, programs can be designed to tailor themselves to the needs of individual students. The key is that progress of the student is stored in a way which is independent of the program being used. Any program can check on the results of previous programs and use them to customize a session for the student.

5 Summary

The major advantage of the new *Layered Multimedia Data Model* (LMDM) is its ability to address within a single modeling framework two issues that make today's multimedia computing difficult: (1) interface between the data and application development; and (2) data sharing and application portability. A clean and efficient modeling paradigm, LMDM conveniently integrates MM data with MM authoring into a unified applications development environment. As a system that follows the computer-integrated multimedia (CIMM) mode, LMDM can be helpful in development of sophisticated, flexible, portable and reusable educational multimedia products. It provides extended capabilities for various interaction and teaching styles, scoring and record keeping, persistent navigation, for student performance evaluation, etc.

References

[Barker, J. and Tucker, R. N., 1990] Barker, J. and Tucker, R. N., editors (1990). *The Interactive Learning Revolution: Multimedia in Education and Training.* Kogan Page, Ltd. (UK).

[Hoepner, P., 1992] Hoepner, P. (1992). Synchronizing the presentation of multimedia objects. *Computer Communications*, pages 557–564.

[Klas, W. et al., 1990] Klas, W., Neuhold, E., and Schrefl, M. (1990). Using an object oriented approach to model multimedia data. *Computer Communications*, pages 204–216.

[Lipton, R., 1992] Lipton, R. (1992). *Multimedia Toolkit: Build Your Own Solutions with DocuSource.* Random House Electronic Publishing.

[Pea, R. D., 1991] Pea, R. D. (1991). Learning through multimedia. *IEEE Computer Graphics and Applications*, pages 58–66.

[Schloss, G. A. and Wynblatt, M. J., 1993] Schloss, G. A. and Wynblatt, M. J. (1993). A layered model for multimedia data. In *Proceedings of the 3-rd Int. Workshop on Information Technologies and Systems (WITS-93)*, pages 142–151.

[Skill Dynamics, 1993] Skill Dynamics (1993). Introducing the IBM interactive multimedia classroom. Videotape LV31-3567-00, Skill Dymnamics, An IBM Company.

[Ward, P. S. and Arshad, F. N., 1991] Ward, P. S. and Arshad, F. N. (1991). Interactive multimedia information systems for education. In *Proceedings of the 1-st Eurographics Workshop on Multimedia*, pages 337–341.

[Woelk, D. et al., 1986] Woelk, D., Kim, W., and Luther, W. (1986). An object-oriented approach to multimedia databases. In *Proceedings of the ACM-SIGMOD Conference on Management of Data*, pages 311–325.

New Aspects of a Hypermedia University Representation

H. MAURER and A. SCHNEIDER
Hypermedia Unit
University of Auckland, Auckland, New Zealand
E-Mail: {H_Maurer, A_Schneider}@cs.auckland.ac.nz

Abstract: Currently, a hypermedia based UTICS (University Transaction, Information and Communication System) is under development at the University of Auckland. This multi-platform system will provide alternatives to traditional forms of information dissemination and public relations exercises. In this paper we report on various components of the UTICS. A virtual walk-through or fly-through will offer users a special way of exploring the university campus. By coupling the hypermedia system with an expert system, the UTICS will support students in their process of course selection. Finally, we show how a UTICS can assist people from outside in accessing expertise sourced at the university.

Introduction

A university transaction, information and communication system (UTICS) is an important and ambitious hypermedia system which aims to provide information about a university as a whole, communication between the various parts of a university and transaction based facilities for intra-university affairs. Universities are complex organizations because of the large number of people, the variety of organizational entities such as service providers, facilities, departments and physical entities such as rooms, buildings and general facilities which are associated by coordinated activities. Because the components of a university serve so many different purposes, a UTICS which is intended to represent the whole structure of a university in a hypermedia-based system has to contain a considerable amount of data and must offer a variety of information services.

A UTICS is supposed to help entities at the university maintain and exchange documents (letters, applications, articles) and to help in retrieval of information (e.g. finding publications, administrative documents, phone numbers or books in libraries). It should minimize and simplify the communication between individuals and groups at the university (by providing email, discussion boards, advertisements), support the instructional process (teaching, learning and training), aid in administrative tasks (scheduling courses, managing rooms), help with transactional processes (crediting and debiting of fees, automatic shipment and distribution of paper documents) and it should provide a means of communicating information to other institutions and to the public (e.g. available information should include degree prescriptions and study programmes, public events, technical resources and staff expertise for industry, government, media representatives and current or prospective students).

At the University of Auckland such a UTICS is under development. The entire system will be implemented with the networked hypermedia system Hyper-G (Kappe, Maurer & Sherbakov, 1993). Parts of the system will also be available in a single computer version distributed on a portable harddisk or CD-ROM/CD-I disks. The latter version will be restricted in that it will contain only those components of the system that serve public relation functions.

In this article we focus on two specific domains of future UTICS applications at the University of Auckland. The first part of this paper deals with a walk- or fly-through of the university campus. We discuss several questions regarding this virtual tour.

In the second part we deal with how university based information can be disseminated to reach specific groups of users. We propose a consultation component in the UTICS based on an expert system that offers information and advice about educational programmes at the university. Finally, we describe how the UTICS will assist public media, government or industry representatives in accessing expertise and equipment at the university.

Walk-through and Fly-through

Introductory remarks

For parties interested in the university, a trip through the university campus is a good introduction. The UTICS of the University of Auckland will offer a virtual journey through the campus. At the moment, there are two techniques for incorporating a trip into a hypermedia system.

The less complex one is the "walk-through", which consists of virtual walks on a given network of paths through the university campus. Attached to every path or place is some multimedia item like a picture or a video clip, taken from the real world, that is displayed as the user reaches each new section of the trip.

The more complex technique is a "fly-through" (Pimentel & Teixeira, 1993) (Earnshaw, Gigante & Jones, 1993). Here, movement is not restricted to given paths and places. A fly-through is based on a 3D model, e.g., derived from a CAD system model.

With both kinds of virtual trip through an environment a number of problems related to navigation and orientation have to be dealt with. These are determined by the technical restrictions of current computer hardware as well as by conditions of human perception and power of imagination.

Current computer graphics hardware typically includes a high-performance graphics rendering unit, a high-resolution colour display, a keyboard and some pointing and dragging device. In most cases the pointing and dragging device is still a mouse, although other devices like joysticks, spaceballs or data gloves offer more degrees of freedom and better control.

Moving and viewing inside a virtual world is an unusual environment for a person's normal senses of navigation and orientation. The situation in front of a computer screen is very poor in information diversity, compared to the real world where humans have a rich variety of senses to help them navigate and orientate themselves within their environment. In the virtual world, quite complex movements may be necessary or desirable; a user's only way to follow a course is through a window that allows a view into the virtual world.

It is therefore a complex and demanding task to design navigation and orientation support tools for a user-interface which does not require any kind of training before actual use (Shneiderman, 1992).

Comprehension of the current position and direction of movement in the virtual world as well as the current viewing direction must be as simple as possible and the screen layout must not be overcrowded with information. The support information should be as far as possible in the "background". For instance, a mouse click to pop up orientation information or a glimpse to the corner of the screen should be sufficient to find one's bearing.

The only way to satisfy these requirements is to match technical and human resources. A clever combination and exploitation of existing visual presentation tools with which users are already acquainted (e.g. maps, arrows and compasses) can provide the means for a solution to this problem.

Walk-through

In a walk-through, the user travels a predefined network of virtual paths. When arriving at a branch in the virtual path network, the user can choose which way to go next. The user can also return to a place already visited and follow another path if they choose. Depending on the multimedia items that are linked to certain stages of a walk, various actions can be taken. In the case of a picture a user could look at some part more thoroughly by zooming in. A video clip could, for example, be watched at greater speed or even backwards.

Even though a walk-through is restricted to a predefined network of trails the user can easily become confused by the continuous presentation of images and video clips. A sequence like ⟨ video clip of a path → selection of a path which continues in a perpendicular direction → video clip of the chosen path ⟩ may look as if the two paths go in the same direction as the usual perception of a rotational motion is lacking. If users do not get any additional information about the layout of the path network, i.e., if there is no overall orientation information provided, they will lose their bearing.

A possible solution for this problem is to provide a map that shows the user's current position in the network (cf. Figure 1). However, only indicating the current position and movement direction is not enough. In addition to this, the map should show the track the user has followed. It would also be useful to show the relative scale between paths already followed and the length of the paths as a whole.

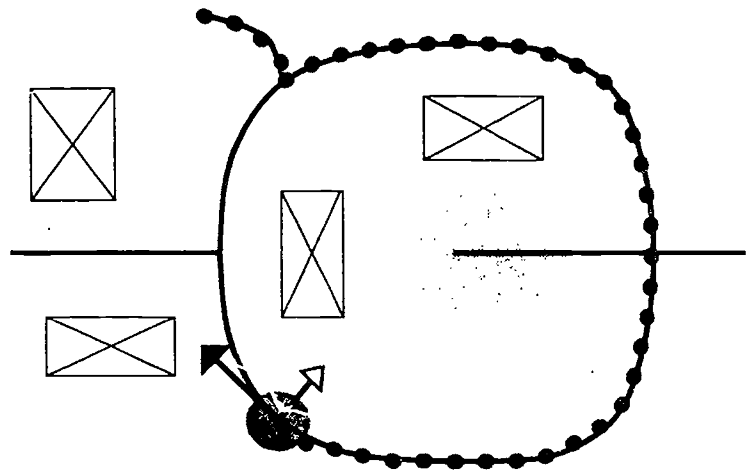


Figure 1. Network trails in a walk-through including marked motion trace, position of user, direction of movement and direction of viewing

Such a map is an additional tool that supports users in the walk-through: it may not take much space on the screen. As with large images that cannot be fully displayed on one computer screen, the presentation system must offer a way to move a window over a map and to zoom in and out so that every part of the map can be seen.

Of course, the panning speed of the window across the map needs to be adjusted to the current level of zoom. The more the user has zoomed in the slower the panning motion has to be and vice versa.

Similar aspects have to be taken into account with panoramic images which constitute a new type of 2D image in multimedia. Whereas still images are bounded in both dimensions, panoramic images are not bounded in the horizontal dimension. They depict a 360° view from a single viewing location by rotating the viewing direction.

There are two different ways to assemble the pictorial material for this type of image. One is to take a series of photographs in different directions. The other is to take a video camera and record the views while rotating the camera around a vertical axis. In both cases, the result is a series of pictures that can be joined to form a 360° view.

Previously, such a large amount of pictorial material could only be displayed by a videodisc which can store and quickly present possible viewing directions. For the UTICS of the University of Auckland, panoramic images will be created by "gluing" pictures of different angles into one 360° image (Hill, 1990) (Foley & van Damm, 1990) (Dix, Finlay, Abowd & Beale, 1993). In order to get a better ratio between the vertical and horizontal extent of the image, images will be taken in the same horizontal direction but with different vertical angles. These will be glued above each other, forming multitrack circular bands.

Displaying panoramas in a walk-through should use the same scene controls for panning and zooming that are offered for bounded maps or images. Because of the circularity of the panoramic image an indication of the current viewing angle needs to be present.

Other problems arise with the use of video clips. Motion of pictures adds another level of complexity. Video clips are mapped onto the screen by successive display of single still images. When succeeding images vary slightly from each other the impression of motion arises.

Playing a video clip at a faster speed (a common option in walk-through presentation systems) may cause strange effects. With currently available computer graphics hardware, accelerating the speed of a video clip is achieved by increasing the frame rate, i.e., by omitting a certain ratio of frames from the video clip.

These omissions can make the motions of objects in the scene look odd or even bewildering. A tree with branches swaying in the wind could appear to be trembling or moving in a clipped manner. Increasing the frame rate to make the movement appear faster works fine with global variations between frames but the quality of the presentation is poor if minor motions exist in the scene, like the waving of tree branches.

One solution to this problem would be to cut the respective object out of the image and overlay it as a sprite which is played with a lower frame rate. However, further problems like transitions at the border between the main image and the sprite may occur or other effects of different speeds in the picture may arise.

Fly-through

In contrast to a walk-through with a predefined network of paths and a fixed set of available images and video clips, the virtual world in a fly-through is based on a world modelled in 3D. The user plays the role of a viewer who is able to move freely in the modelled world and to look in any direction and at any detail. In the case of a virtual university campus, this means that the user can visit every building on campus, enter and leave it as often as desired. Understandably, the challenge of free motion and viewing conditions raises navigational and orientational problems for the user in the model.

One exemplary problem concerning orientation is the following: the user has moved close to a wall or even into a wall and the rendering system displays an unstructured or dark area that represents the user's current view. The usual first-aid measures, like turning or zooming, hardly offer any good orientational help.

In the following, we propose a collection of tools that provide the voyager in a fly-through with sufficient orientational and navigational help. The collection of help tools consists of four windows, arranged in a three level hierarchy.

The single topmost window offers the most often used information and interactive tools to the user. The user has the opportunity to modify the speed of movement and to proceed with one or several more steps in the current direction. From this window the two windows on the second level can be called to get more detailed orientational help.

The first of these displays an aerial view of the current scenery which includes the position of the user in the model and a schematic sketch of the user's current surroundings. As with the orientation map in the walk-through, a trace of the user's course in the model is shown in the sketch. The other middle level window shows a side view of the current situation. In that way, the user gets a good idea of the current vertical position in the modelled world.

Common to both windows is the sensitivity of the mouse cursor's shape to the sort of graphical item or location in the model sketch on which the mouse cursor is sitting. Every shape stands for the sort of action that can be initiated at that position. A brief textual description of the respective action is shown in a small text field within the window. The most extensive use of the mouse cursor's shape sensitivity is made inside the two schematic sketches of the model where it either means "jump to that location in the model" or "display the parameter control window".

This in turn is the lowest in the hierarchy of interaction tools. It offers a complete set of graphical interactive items which give detailed navigational and orientational information relating to the user's current position.

Here, projections of the moving and the viewing direction onto a horizontal plane are shown with compass-like items. A needle can show a viewing direction or a cone can depict the whole viewing angle, with its opening angle dependent on the extent of zooming.

Provision of information

Expert system assisted consultation

Prospective students who intend to take up a study programme at the university form one group of persons with their own information requirements. Their situation is difficult as they are confronted with a wide range of choices between degrees. In order to make the right decision they have to consider a large amount of information from several sources. There are university calendars which tell about the courses being offered, regulations for admission and enrolment procedures. Specific material informs about academic contact people in faculties and departments, about available options for financing education, about accommodation places and about career advisory services. All of this is rather overwhelming for students who have not had contact with a university before. Help is clearly needed for this decision-making process.

To overcome this difficulty we propose extending the hypermedia system to include an expert system. Expert system techniques have been used for many purposes in the last few years and they appear to be an appropriate way of dealing with the decision making.

An expert system which contains declarative and procedural knowledge about the problem field can play the role of a counsellor by assisting students to lay out their course options. The student advice situation is not unique, however. As the amount of available information in hypermedia systems grows, there will be an increasing need for such means to assist users with information retrieval.

By introducing methods of knowledge representation and automated reasoning from the area of artificial intelligence, hypermedia becomes "intelligent" (Woodhead, 1991). The raw information stored in a hypermedia system becomes knowledge that is structured and accessible in a way required by special applications and by users.

Technically speaking, the integration of an expert system into a hypermedia database means using the expert system as a subsystem of the hypermedia system. During a user's normal interaction with the hypermedia system the actions taken by the user are recorded and observed by the expert system. By collecting data about the user it develops a profile of the user. This user model in turn can be employed to influence the kind of node information and the number of links which are presented to the user in later stages of the navigational process. In that way, the user's search space in the hypermedia system can be reduced and the information available can be tailored specifically for the user's needs.

With a hypermedia system containing an expert system component, the onus of decision-making, in a dynamic context, is more with the system than with the user. This means an emphasis on context-sensitive guidance, as opposed to undirected navigation or browsing.

The main problem with the combination of a hypermedia system and an expert system is finding a way to provide the information that an expert system needs to function in a suitable knowledge representation form. In addition to this, the results of the reasoning steps performed by the system have to be usable by the hypermedia system, i.e., the presentation part of the coupled system must be able to adjust the type of information and the number of links offered to the user, according to the current parameters given by the expert system subunit.

Access to university expertise

Another application of a hypermedia-based UTICS is the provision of information about university sourced expertise and equipment available to the public, or groups from media, industry and government. A few years ago, at the University of New York, USA, a system called ProfNet was established which put the media in touch with academic specialists. Recently, a similar system named Xpertnet has been introduced at Macquarie University, Australia (Richardson, 1993a) (Richardson, 1993b).

Both of these systems receive queries from journalists by fax or email and send them on a daily bulletin to cooperating universities. These in turn provide journalists with contact details for academics who can answer their queries or provide suitable comment. In this way, journalists can get in contact with specialists all over their country more quickly and easily.

However, the establishment of contact between a journalist and an academic still takes several hours and staff at universities are often unable to match the incoming query with areas of expertise. In this case their only way of finding a match is to contact the journalist or over-answer the query by providing every possible reply. This scenario then requires effort on the part of the journalist who must choose from the list of answers.

Moreover, ProfNet and Xpertnet are only intended for media representatives. There are other persons and groups outside the academic domain who are interested in university resources.

System like ProfNet and Xpertnet that are based on networked hypermedia systems can do better. If the information about university staff members and their areas of expertise together with information about departments and their equipment is recorded in a UTICS instead of only being available in paper form like a media contact directory or a research report, then the information can easily be kept up to date. Interested persons from outside the university can connect to the system and search inside it on their own. With comprehensive information about staff and equipment users can make use of their existing knowledge in order to find a match for their query.

Conclusion

Altogether the UTICS project at the University of Auckland will span several years. The entire project will need a work force of several academics, programmers and technical staff. There will be many more components than those discussed above included in the system (Lennon & Maurer, 1993a) (Lennon & Maurer, 1993b).

There are still, however, a multitude of problems to solve. For example, panoramic images, require a reliable procedure for "gluing" single images together. At the moment algorithms provide semi-automated morphing procedures. The designer identifies several points in different images that have to be matched and the software carries out the matching for all of the points between. The next step in this direction will be more sophisticated programs which automatically find corresponding points and glue images together without human assistance.

Quite complex problems remain with walk- and fly-through systems. How detailed do orientation maps of the virtual world need to be? Or, which floor is shown in a side view schematic sketch when the user is between two floors of a virtual building? As a first approach to the virtual fly-through, the 3D scene software included in the hypermedia browsing system Harmony (Andrews, Kappe & Schipflinger, 1994) has been used to develop a small model called the "Virtual University of Auckland".

References

Andrews, K., Kappe, F., & Schipflinger, J. (1994, June). Harmony: A tool for navigating through deep hyperspace. In *Educational Multimedia and Hypermedia Annual*, Vancouver. ED-MEDIA 94, AACE Publishing.

Dix, A., Finlay, J., Abowd, G., & Beale, R. (1993). *Human Computer Interaction*. Prentice Hall, New York.

Earnshaw, R. A., Gigante, M. A., & Jones, H. (Eds.) (1993). *Virtual Reality Systems*. Academic Press, London.

Foley, J. D. & van Damm, A. (1990). *Computer Graphics: Principles and Practice*. Addison-Wesley.

Hill, F. S. (1990). *Computer Graphics*. Macmillan, New York.

Kappe, F., Maurer, H., & Sherbakov, N. (1993). Hyper-G - A Universal Hypermedia System. *Journal of Educational Multimedia and Hypermedia*, 2(1):39 - 66.

Lennon, J. & Maurer, H. (1993a, July). Lecturing technology: A future with hypermedia. Technical Report 74, University of Auckland, Auckland.

Lennon, J. & Maurer, H. (1993b, September). Applications of hypermedia in universities. Technical Report 78, University of Auckland, Auckland.

Pimentel, K. & Teixeira, K. (1993). *Virtual Reality - Through the new looking glass*. Mc Graw Hill, New York.

Richardson, J. (1993a, August). Net to catch profs for the media. *Campus Review*, 3(29).

Richardson, J. (1993b, August). Uni news network puts journalists in picture. *Campus Review*, 3(31).

Shneiderman, B. (1992). *Designing the user interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley, Reading, MA, second edition.

Woodhead, N. (1991). *Hypertext and Hypermedia - Theory and Applications*. Sigma Press, Wilmslaw.

Acknowledgements

Thanks to Paul Hulford and Michael Lennon for their invaluable assistance with English during the composition of this paper.

Multimedia Design Principles for Constructing Prescriptive, Democratic and Cybernetic Learning Environments

RICHARD A. SCHWIER
Department of Curriculum Studies
University of Saskatchewan, Saskatoon, Saskatchewan S7N 0W0, Canada
E-Mail: schwier@sask.usask.ca

Abstract: This paper explores principles for designing interactive multimedia instruction based on a classification scheme for multimedia environments which includes prescriptive, democratic and cybernetic categories. If a multimedia developer first considers the nature of the desired learning environment, a continuum of instructional design decisions naturally obtain. This paper considers several design issues, including control, practice, feedback, cooperation, and metacognition, and provides examples of how they are expressed in different multimedia environments.

The quality of multimedia-based interaction is more the product of the way instruction is designed, and less the result of the system on which it is delivered. To fully exploit the capabilities of more powerful instructional technologies, designers must also reexamine the assumptions and expand the strategies we employ in instructional design . This paper abbreviates several ideas which due to space restrictions, cannot be elaborated fully here. For a more thorough treatment of this topic, I encourage you to contact me directly, and I will send you a complete paper.

Three Learning Environments

At least three distinct learning environments have been identified for individualized instruction and subsequently adapted to interactive multimedia learning (Romiszowski, 1986; Schwier, 1993). These include prescriptive, democratic and cybernetic learning environments. Prescriptive environments specify learning objectives and the instructional system is used as a primary delivery medium. In most cases the boundaries of learning are externally imposed, and the learner's role is to receive and master the given content.. Democratic environments emphasize the learner's role in defining what is learned, how it is learned, and the sequence in which it is learned. Democratic learning resources emphasize navigation, motivation and access, and they down-play objectives and evaluation. Cybernetic environments emphasize a complete system in which the learner can interact freely and naturally with the instruction, which in turn responds dynamically with the learner. The cybernetic instructional environment, unlike instruction provided in prescriptive and democratic environments, actually expands beyond the initial design decisions made during its development. This expansion marks its difference from being merely a sophisticated prescriptive environment; the very substance of the learning landscape is changed by the nature of interactions during instruction, not just the path followed by an individual through existing material (whether prescribed or democratic).

This paper applies five instructional design issues to each of these environments— control, practice, feedback, cooperation, and metacognition. How might each of these notions be expressed in different multimedia environments?

Instructional Design Principles for Prescriptive Environments

Prescriptive Learner Control

In prescriptive environments learner control speaks to when learners might be empowered by being given more control over instruction and conversely when learners might be hampered by having such control. In other words, a prescriptive environment assumes the question, "should learners be given or denied control of learning?" In designing prescriptive multimedia environments, control is often expressed in the selection of content and sequence. Naive or uninformed learners require structure, interaction, and feedback to perform optimally (Carrier and Jonassen, 1988; Higginbotham-Wheat, 1990; Kinzie, Sullivan, and Berdel, 1990). Giving ill-prepared learners control over instruction may permit them to make poor decisions about which content is important and how much practice is required, which may in turn be reflected in decremented performance (Coldevin, Tovar, and Brauer, 1993)

Example: A pre-test is given at the beginning of a tutorial program. Those learners who score higher on the pre-test are informed of objectives they are expected to achieve, and placed in less restrictive versions of the tutorial. Those who score lower are placed in highly prescribed, remedial programs.

Prescriptive Practice

Practice events in prescriptive instruction should require learners to use information; that is, practice during prescriptive instruction emphasizes acquisition of specified skills and knowledge. In prescriptive environments, practice should be imposed often during early stages of learning and less often as time with a particular topic progresses (Salisbury, Richards, & Klein, 1985). As facility and familiarity with the learning task increase, so should the difficulty of practice. In prescriptive multimedia environments, the difficulty level would be managed externally by giving the learners access to progressively more difficult tasks only as they are successful with previous material.

Example: Embed several relatively easy questions in the first sections of a program. When the learner demonstrates mastery of simple questions on a particular topic, give fewer, but progressively more difficult questions.

Prescriptive Feedback

In prescriptive environments, feedback will often take the form of error detection and correction. Because instructional learning outcomes are explicit, it is possible (and probably desirable) to funnel the feedback about learner performance toward the intended outcome. Feedback acts to correct errors, and the effect is more powerful if the learner feels confident that the incorrect response is correct, yet in verbal information tasks, correct response feedback is better than no feedback. Instruction utilizing response certitude estimates is less efficient, probably due to the time taken by the learner to make estimates, but efficiency is seldom a major consideration in interactive multimedia instruction. Feedback effects also tend to be stronger when no other instructional text is present, thus increasing the feedback's informational effect.

Example: Have learners declare their level of confidence at key points during instruction, and then tailor feedback to acknowledge the professed level of confidence.

Prescriptive Cooperative Learning Strategies

In prescriptive environments, learners use cooperative learning strategies to collectively address externally defined tasks. The goal is to bring a broad range of skills and collective energy to bear on a complex task. One strategy is to emphasize interdependence and accountability: The performance of each member of a cooperative group must contribute to the group's achievement, and the reward structure must account for this. Another strategy is promotive interaction, where an individual's effort to bolster the efforts of other group members is promoted through heterogeneous grouping. Yet another issue is training for collaboration. Learners can be trained to use interactive strategies effectively, and training should be both content-specific and content-independent (Hooper, 1992).

Example: Multimedia materials may include an independent module for developing global collaborative skills, and providing specific strategies to be used within the program by the group.

Prescriptive Metacognitive Strategies

In prescriptive environments, metacognitive skills are used by the learner to notice when something is not clearly understood and take appropriate remedial action. They may also include selecting from among various memory strategies, including: "metamemory" skills with different memory strategies and an awareness of how to choose which is appropriate for a given task; and "metacomprehension" skills which include being able to detect when one fails to comprehend something and is able to take appropriate remedial action.

Example: Encourage learners to watch for difficulties, and think about ways to find answers to their problems.

Instructional Design Principles for Democratic Environments

Democratic Control

From a democratic perspective, learner control is assumed. It is central assumption in a democratic environment that learners construct multiple—and equally valuable—realities from their unique interactions with instruction. If the learner is to be given control over significant decisions, it is incumbent that they also be given the skills necessary to exercise the control. Multimedia designers cannot assume that learners are able to assume control without learning *how* to take control and make productive decisions. Several applicable design principles can be extracted from the literature. Learners who are generally high achievers or who are knowledgeable about an area of study can benefit from a high degree of learner control (Hannafin and Colamaio, 1987). Giving the learner control may increase motivation to learn, but it does not necessarily increase achievement and may increase time spent learning (Santiago and Okey, 1990).

Example: Offer learners control of a program. Let them select topics and sequence, but provide advice when appropriate about the possible consequences of their choices.

Democratic Practice

In democratic environments, practice may include use and review of specific content or skills, but also includes practice with strategies for learning, not just practice with specific content or skills. Learners can benefit from memory and organizational strategies to make information more meaningful. Practice events in democratic multimedia environments should encourage learners to use information and discover and derive new relationships in information. Democratic multimedia instruction should be designed to include opportunities for learners to acquire strategies for using the instruction optimally. Instructional cues can be used productively, even with passive learners, to promote exposure to elaborations, and consequently increase time-on-task and achievement (Lee and Lehman, 1993). Practice in democratic and cybernetic environments should be varied and available to the learner at any time, and in several forms to satisfy self-determined needs.

Example: Include "strategy" modules within a program to apply newly learned content to novel situations, or provide on-line advisement to learners reminding them to practice using new content.

Democratic Feedback

In democratic multimedia environments, feedback will often take the form of advisement; that is, informing learners about the consequences of choices and patterns of choices, or providing recommendations about productive avenues of study. Because learners are defining specific learning outcomes in democratic environments, feedback will attempt to help learners articulate their own goals and help them follow productive paths through the learning system. Learners maintain or change their own cognitive operations based on how new information about their performance matches their expectations about performance (Mory, 1992). Delayed feedback may be more effective for higher cognitive tasks than immediate feedback. It is possible that delaying feedback allows additional time for reflection, and this in turn may facilitate learning challenging material.

Example: Include an on-screen "advise" button for the learner. When selected, it provides the learner with a visual map of their choices and paths (audit trail) and highlights preferred alternatives for the learner with explanations.

Democratic Cooperative Learning Strategies

In a democratic environment, cooperative learning strategies place emphasis on teamwork and shared responsibility for decision-making. Some appropriate strategies include training for collaboration. Learners can be trained to use interactive strategies effectively, and training should be both content-specific and content-independent (Hooper, 1992). Multimedia materials may need to include an independent module for developing global collaborative skills. Debriefing sessions following group processes should also be included to allow group members to reflect on effective and ineffective strategies they used.

Example: Have learners input the number of users for materials, and provide alternative designs. Each member of a team is given specific problems to solve or activities to complete during the program, and the program monitors individual, as well as group, progress toward completion of the instruction.

Democratic Metacognitive Strategies

In democratic multimedia learning environments, the issue becomes one of not leaving the learner adrift in a sea of content without the tools to be successful, and increasing the metacognitive demands placed on the learner (Park and Hannafin, 1993)."Self regulation" is an individual's ability to make fine adjustments to errors detected when the instruction provides no feedback. "Schema training" has to do with getting the learner to generate personally relevant structures for understanding material, and becoming less dependent on structures provided by the instruction. "Transfer" is the ability of the learner to apply a strategy to an unfamiliar and dissimilar learning task. Metacognitive strategies can promote learning and can be generalized across learning situations, but they must be learned and practiced. In any instruction, but particularly in democratic and cybernetic multimedia environments, metacognitive strategies can be learned which will help the learner make more productive decisions (Osman and Hannafin, 1992).

Example: One method of developing such strategies is learner advisement about metacognitive strategies. Learners can be given reminders about ways to approach materials (e.g. "Have you thought about trying this approach? It worked for you the last time you tried it."). The focus with such strategies should be on providing metacognitive prompts and promoting self-generated strategies, while weaning the learner from prompts as quickly as possible.

Instructional Design Principles for Cybernetic Environments

Cybernetic Control

Control is negotiated in cybernetic multimedia environments, and is the product of mutual exchange. The learner may be advised about difficulty levels and productive choices, but decisions will be left in the hands of the learner. Decisions result in consequences to which the learner must respond. Learner control with advisement seems to be superior to unstructured learner control for enhancing achievement and curiosity, promoting time-on-task, and stimulating self-challenge (Arnone and Grabowski, 1991; Mattoon, Klein, and Thurman, 1991; Milheim and Azbell, 1988; Santiago and Okey, 1990). Further, the amount of control available to a learner at any particular time in a program should not necessarily be fixed. Courseware should be adaptive. It should be able to alter instruction dynamically, based on learner idiosyncrasies (Carrier and Jonassen, 1988).

Example: Learners enter a fully simulated landscape of the human body and navigate by pointing in the direction they want to move. Navigation options for making choices increase as the learner navigates successfully.

Cybernetic Practice

The purpose of practice in a cybernetic environment subsumes using information, applying information, and testing new skills and information, but all are done in a simulated and saturated environment. Practice approximates practicing in a real environment, and the growth in a learner's ability to perform new skills is commensurate with actual skill levels.

Example: A surgeon practices hip replacement surgery in a virtual surgery theater. Learner acquires new skills and practices using them in a full simulation, and is given more challenging situations as skills increase.

Cybernetic Feedback

In cybernetic environments, feedback can be characterized as mutual and negotiated. Learners set directions and make choices, and the learning system "learns" from patterns which emerge how to respond to the learner or provide new challenges. Feedback will often provide a "metacognitive viewpoint" for the learner, responding naturally and logically to learner actions, identifying intentions, and establishing levels of challenge for the learner.

Example: The system identifies a pattern of responses, say a tendency for the learner to respond to oral challenges in French, rather than English. The system provides feedback in French, and monitors whether the learner performs more successfully. If the learner falters, the system switches back to English, and informs the learner of the decisions made.

Cybernetic Cooperative Learning Strategies

In cybernetic environments emphasis may be placed on collaborative problem solving or competition within a stimulus-rich environment. Learners may "join forces" to address problems presented in a simulated environment, and the system will have a wider array of input from which to "learn." In a competitive cybernetic environment, learners may be pitted against one another in a simulation.

Example: Two participants are given the task of designing the dining areas in a restaurant to obtain optimum seating arrangements. Both are placed in a virtual restaurant, and asked to move tables. One is to concentrate on maximizing workflow for employees, the other is concerned with seating as many diners as possible. As they make decisions, the system develops rules for optimizing seating designs.

Cybernetic Metacognitive Strategies

In cybernetic environments, systems "tune" themselves to the metacognitive strategies employed by learners, adjust to them, and advise the learner of trends which emerge. This assumes that programs are sufficiently sophisticated to extrapolate meaningful trends from patterns of learner responses, a type of cybernetic metacognition not readily available. Osman and Hannafin (1992) warn against designs in which the training in metacognitive strategies require more energy than the content to be learned. The lesson: beware metacognitive overhead costs.

Example: In a virtual restaurant environment, learners are given the ambiguous task of designing dining areas. As learners impose designs on seating arrangements, the system monitors decisions made, and extrapolates the metacognitive strategies employed (e.g. one learner is designing for maximum seating possible, while another is attempting to create an aesthetic and intimate atmosphere).

References

Alessi, S.M., & Trollip, S.R. (1985). *Computer-based instruction: Methods and development.* Englewood Cliffs, NJ: Prentice-Hall.

Arnone, M. P., & Grabowski, B. L. (1991). Effect of variations in learner control on childrens' curiosity and learning from interactive video. In M. R. Simonson and C. Hargrave (Eds.), *Proceedings of the 1991 Convention of the Association for Educational Communications and Technology* (pp. 45–67). Orlando, FL: Association for Educational Communications and Technology.

Beasley, R.E. (1992). A methodology for collecting navigation data using IBM's *LinkWay:* A hypermedia authoring language. *Journal of Educational Multimedia and Hypermedia, 1*(4), 465-470.

Carrier, C. A., & Jonassen, D. H. (1988). Adapting courseware to accommodate individual differences. In D. H. Jonassen (Ed.), *Instructional designs for microcomputer courseware.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Coldevin, G., Tovar, M., & Brauer, A. (1993). Influence of instructional control and learner characteristics on factual recall and procedural learning from interactive video. *Canadian Journal of Educational Communication, 22*(1), 113-130.

Hannafin, M. J., & Colamaio, M. E. (1987). The effects of variations in lesson control and practice on learning from interactive video. *Educational Communications and Technology Journal, 35*(4), 203–212.

Higginbotham-Wheat, N. (1990). Learner control: When does it work? In M. R. Simonson and C. Hargrave (Eds.), *Proceedings of the 1990 Convention of the Association for Educational Communications and Technology.* Anaheim, CA: Association for Educational Communications and Technology. (ERIC Document Reproduction Service No. ED 323 930)

Hooper, S. (1992). Cooperative learning and computer-based instruction. *Educational Technology Research and Development, 40*(3), 21-38.

Kinzie, M. B., Sullivan, H. J., & Berdel, R. L. (1988). Learner control and achievement in science computer-assisted instruction. *Journal of Educational Psychology, 80*(3), 299–303.

Lee, Y.B., & Lehman, J.D. (1993). Instructional cuing in hypermedia: A study with active and passive learners. *Journal of Educational Multimedia and Hypermedia, 2*(1), 25-37.

Mattoon, J. S., Klein, J. D., & Thurman, R. A. (1991). Learner control versus computer control in instructional simulation. In M. R. Simonson and C. Hargrave (Eds.), *Proceedings of the 1991 Convention of the Association for Educational Communications and Technology* (pp. 481–498). Orlando, FL: Association for Educational Communications and Technology.

Milheim, W. D., & Azbell, J. W. (1988). How past research on learner control can aid in the design of interactive video materials. In M. R. Simonson and J. K. Frederick (Eds.), *Proceedings of the 1988 Convention of the Association for Educational Communications and Technology* (pp. 459–472). New Orleans, LA: Association for Educational Communications and Technology. (ERIC Document Reproduction Service No. ED 295 652)

Mory, E.H. (1992). The use of informational feedback in instruction: Implications for future research. *Educational Technology Research and Development, 40*(3), 5-20.

Osman, M.E., & Hannafin, M.J. (1992). Metacognition research and theory: Analysis and implications for instructional design. *Educational Technology Research and Development, 40* (2), 83-99

Park, I., & Hannafin, M.J. (1993). Empirically-based guidelines for the design of interactive multimedia. *Educational Technology Research and Development, 41*(3), 63-85.

Romiszowski, A.J. (1986). *Developing auto-instructional materials.* New York: Nichols Publishing.

Salisbury, D.F., Richards, B.F., & Klein, J.D. (1985). Designing practice: A review of prescriptions and recommendations from instructional design theories. *Journal of Instructional Development, 8*(4), 9-19.

Santiago, R. S., & Okey, J. R. (1990, February). *Sorting out learner control research: Implications for instructional design and development.* Paper presented at the Annual Conference of the Association for Educational Communications and Technology, Anaheim, CA.

Schwier, R.A. (1993). Learning Environments and Interaction for Emerging Technologies: Implications for Learner Control and Practice. *Canadian Journal of Educational Communication, 22*(3), 163-176.

Integrating Concept Networks and HyperMedia

PETER J. SCOTT & DAVID J. ARDRON

Learning Research Group
Department of Psychology, University of Sheffield, POB 603, Sheffield S10 2UR, UK.

This paper reports on an integration of concept mapping for manipulating and structuring knowledge with multimedia presentation technology. It presents the current development of the HyperMedia Activated Network (HAN) authoring environment. HAN is an intelligent hypertext demonstrator system. The first HAN module, called ICADT, was developed initially to teach Computer Aided Architectural Design (CAAD) using the GABLE 4D™ CAAD environment at Sheffield University, (Scott and Lawson, 1990; Scott, Lawson and Ryu, 1991; Scott, 1993). Following a detailed evaluation of the use of this system to teach second year students of Architecture at Sheffield University the authoring and inference aspects of the system were substantially improved. The system is now being extended to demonstrate the support of some aspects of Cognitive Science and Psychology. The HAN system integrates artificial intelligence and flexible learning techniques in a dual-platform experiment with hypertext and semantic network architectures.

The intelligent hypermedia system HAN works as a complete experimental tutorial environment that generates small hypermedia tutorials on-line as they are required. A student query activates a portion of a semantic network representing the key concepts of a domain of a HAN module. The activated portion of the network and a simple model of the student is used by the system to construct a small mapped hypertext which can then be explored by the student. The HAN system combines hypermedia material for the authoring and tutorial interface provided by Apple's HyperCard™ with a core semantic network management system implemented in Logic Programming Associates' MacProlog™. These applications communicate via Apple's Apple Event™ interface. The system is intended to provide a radical hypermedia alternative which aims to address the classic authoring problems faced by Instructional Designers who are experts in their field but not experts in the presentation technology, (Nicolson and Scott 1986).

The HAN Shell

The HAN shell consists of four main sub-systems: A concept network knowledge base including authoring and maintenance; an automatic tutorial generation system; a student-record maintenance system; and a tutorial presentation and performance monitor. This architecture is presented in figure 1.

Within the concept net sub-system, HAN maintains a number of layers of authored material. Authors create hypermedia teaching material about some concept in the domain using HyperCard. This material can be of any internal complexity and can contain for example limited links to glossary items, (but is often just a single screen - or card). This material is termed a "topic" and is seen as an essentially stand-alone chunk of knowledge in some domain. The author then defines the links between the high-level concepts that each topic explains using a simple graphical concept mapping interface. Links can be of various sorts, for example one concept (with its associated topic material) may 'depend-upon' another or be 'an illustration of' another. There are currently eight links in the concept mapping authoring system for Architectural CAD and a similar number for the Language module. The tutorial generation sub-system is responsible for spreading activation though the network of concepts, taking the activated chunk of the network and constructing a tutorial from the associated topic materials. The tutorial generated is then presented to the student by the presentation sub-system. The system manages a simple internal model of each individual user via the record-maintenance sub-system and is able to react to different individuals and different pre-defined user types.

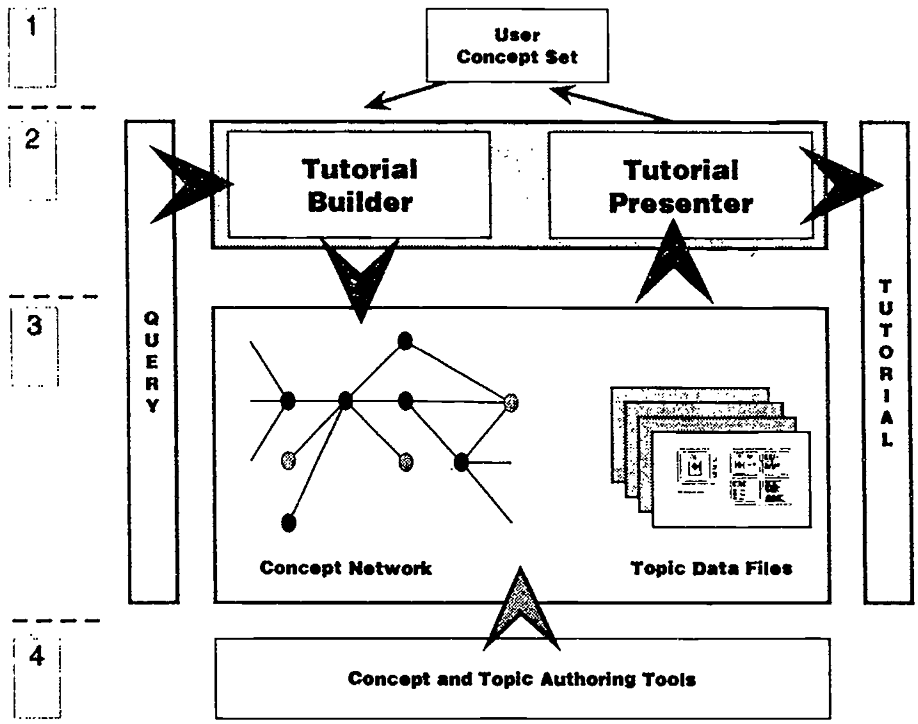


Figure 1. The Architecture of HAN.

In this figure we show that there are essentially four levels at which this system could in principle be used. Level 1 at the top is where the system is related to approaches to expert or intelligent tutoring. In the current HAN this is only supported by relatively primitive user modelling features. Level two is the "query" level where the system can react more specifically to user queries. Level three is the "browse" layer in which most hypertext systems offer users the opportunity to wander through a virtual net of related topics. Finally, level four is the network authoring level in which users could in principle generate their own networks and topics to help structure their understanding. This is the level advocated in concept-mapping systems research. The current work on HAN to date has focused on level two.

In overview then, users typically interact with the HAN system by issuing queries about some concept to be learned. The user's query is mapped onto the HAN concept network. HAN then activates a fragment of the network which it uses to generate an individualised hypertext which it can present to the user with the net fragment acting as a hypermap of the topics involved.

Some detail of the HAN System

(A) The Topic Database. A concept is represented by a "topic" in the ICAD Tutor system, where each topic is the basic (primitive) unit of instruction consisting of several layers. The top layer has a simple graphic representation of the essence of the design topic concept which may be sufficient to explain the core part of the knowledge to be delivered. The basic material can be taken from existing manuals (specific to the CAAD system being used) or other reference materials. In the case of the ICADT module, the topic database was general knowledge about the principles of CAAD system engineered by architects. In the case of the Psychology and Cognitive Science modules the material was collected from lecture and course notes of the individuals supplemented by edited extracts of literature and specially constructed demonstration material. Successive lower layers introduce more detailed information as textual and graphic explanations of the topic using hypertextual links.

(B) The Concept Network Database. Topics are represented at a high level by a knowledge base of 'concepts' each of which is linked to others in a network. The power of the generation system to provide sensitive help to a user resides in an ability to select amongst rich linkages for different and possibly complimentary purposes. The function of this network is to be interpreted by the generation system. The network is a 'virtual' net of single links in Prolog. Each topic has a unique node name in the net and can be associated with other topics by giving a specific link type which represents the semantic relationship between two concepts. Figure 2 illustrates one small fragment of the concept network of the ICADT module of HAN based around the concept of "Offset" in the Computer Assisted Design system. In this case, activation has spread down the directed links to nearby concepts and thence out into the network. The activated part of the network can then be processed by the next part of the HAN system as the basis of the tutorial. The algorithm for the spread of activation is currently very simple and is constrained more by the size of the network itself, the direction of the linkages and the context of the activation (knowledge about the status of the user - novice, intermediate or expert) than by any device for instantiating decay or any other control mechanism. Whilst the presentation mechanisms discussed next could handle any size of network chunk, in practice the network chunks activated are relatively small and can be pruned even further before being presented by knowledge about the individual student. Ongoing research in the Cognitive Science and Language HAN modules is experimenting with a variety of different network pruning heuristics based upon the semantic nature of the network links.

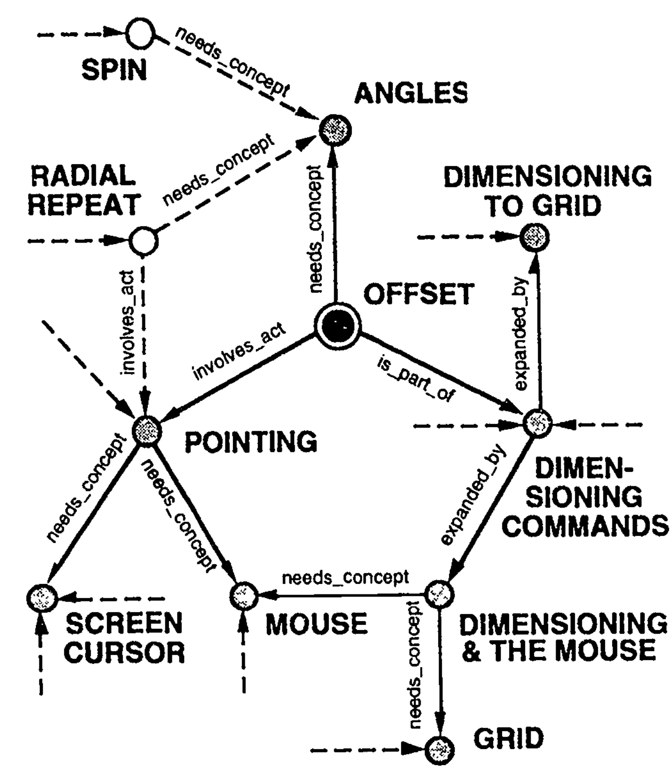


Figure 2. A fragment of the concept network for the Architectural CAD module.

(C) The Performance and User Modelling system. A performance monitor provides individualised record-keeping of each student's use of the system. The tutorial generation system may use this record to "prune-out" material that has already been seen - or to include more complex topic material - which a novice may not be exposed to. To serve for this purpose, it is necessary to keep record of "query topics" and "seen topics" (i.e., what kinds of questions did the users asked to the system and which topics did they studied). In order to make the "Performance Monitor" work, we must ensure that new users log-in to the program (so that the system knows who is using it) and log-out when finished. Logging-in finds the existing user file or creates a new one if necessary and the tutorial presentation system keeps a separate record of the student's query-topic and seen-topic list as it works; and logging out saves an updated version of the user's query-topic and seen-topic list.

(D) The User Interface. The HAN presents an interface to the user that is entirely within the HyperCard point-and-click model. The user need never be aware that the system is sending messages to and from the Prolog inference engine. Figure 3 shows the basic query interface as it is presenting information from the ICADT module of the system. The name of the user (in this instance "Hole") and the user's self selected level ("Beginner") are selected at the top right of the screen. All of the available concepts are listed down the left hand side of the screen for direct manipulation. Alternatively the user can search for keywords (via the upper right hand side window) which may be listed as associated with or paraphrases of single topics. As you would expect, the search can involve simple conjunctions, disjunctions and negations and can be phrased in a simple natural language form. The search results are shown in the lower right-hand side window, where they too can be directly called. Once a query is issued by such a call the results are passed to the Prolog sub-system where the concept net is activated for that user at that level.

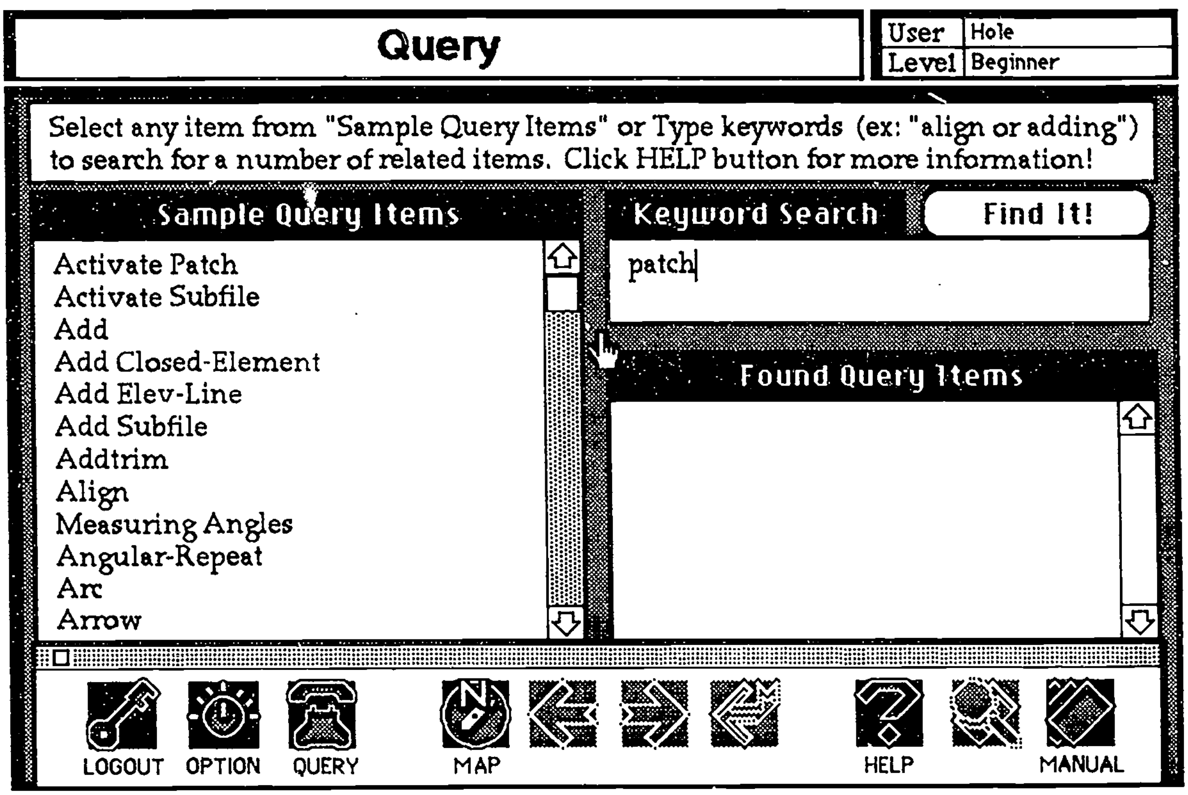


Figure 3. The basic user interface.

The Prolog inference system passes back the activated chunk to the presentation system which uses it as the core "map" of the multimedia material required to understand the chosen query. In figure 4 we see the concept "Align" in the CAD system represented as a simple graphical tree "hypermap" of related concepts leading towards the understanding of the query topic shown to the right hand side of the screen.

In figure 4 we see the user pointing to one of the nodes "Screen" which will then be expanded into the supporting "Topic Unit" for this concept. The topic unit will provide a range of materials to the user to help them understand the topic in question - whilst the map imposes the structure and enables them to explore how each individual topic relates to other concepts connected to this query. As the inference system takes note of individual user's needs and their self-stated rating of expertise, we note that it will produce different maps at different times for the same query.

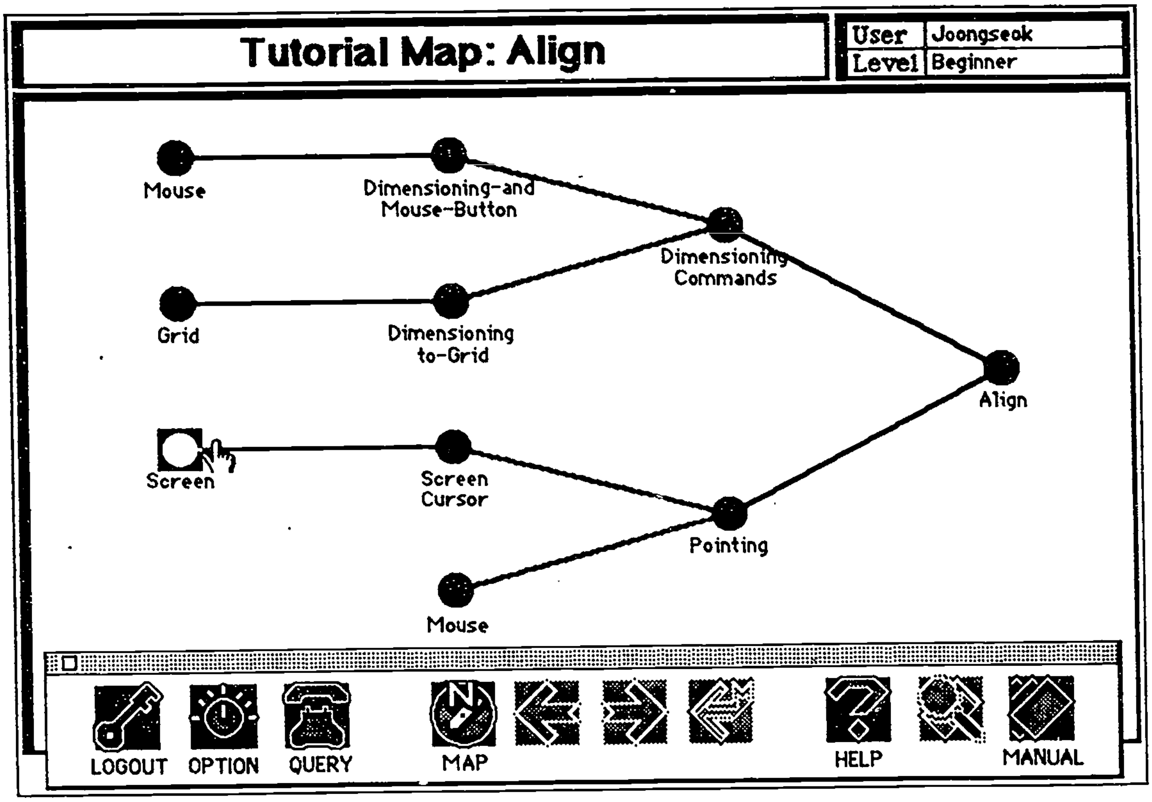


Figure 4. A tutorial map constructed for the concept "Align" for a beginner user.

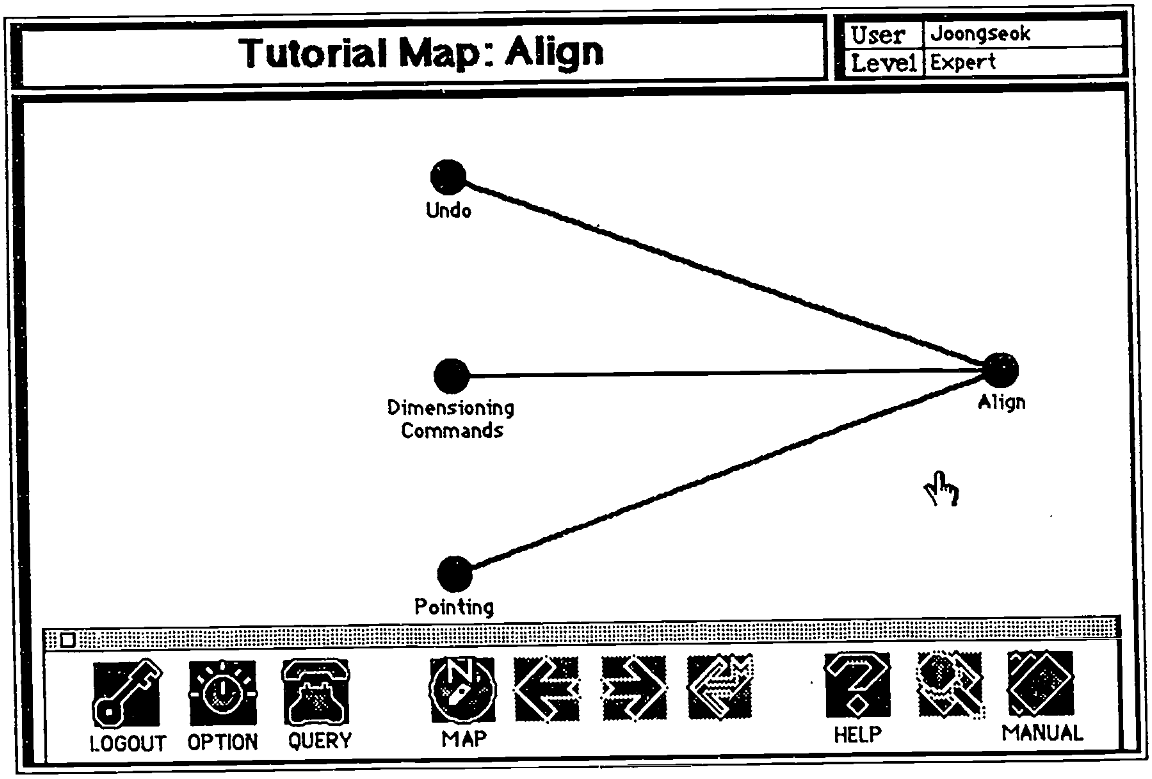


Figure 5. A tutorial map constructed for the concept "Align" for an expert user.

Compared to the identical query of figure 4 it can be seen from figure 5 that the expert user is assumed by the concept network authors to require a higher level overview of this topic - with some basic topics hidden and some extra advanced topics included. This analysis was produced by the authors of the concept network - in this module practising teachers of Architecture at Sheffield University.

Pros, Cons and Development of the HAN System

The HAN system represents a growing trend towards ExperText or ExperMedia in HyperText / MultiMedia research. It authors a HyperMedia "answer" to a student's Query on demand. Its critical strength is that it is inherently based in micro-maps of the knowledge meaning that users don't get lost in hyper space, (Novak 1983). Authored material does not fossilise as easily as traditional hand-crafted computer assisted learning projects. Neither is the system fragile like most Intelligent Tutoring Systems. But most importantly in the current version we have explored how authors can be saved from concentrating primarily on the presentation of their knowledge, but instead enabled to focus on its organisation. The HAN author focuses on the key concepts & their maps.

The HAN system has many technical limitations. The current version is still exploring techniques for using the link types in the network for tutoring reasoning in the new modules. It uses a very primitive user-model, in contrast to the focus of this sort of work within Intelligent Tutoring Systems projects (Wenger 1987). Furthermore it has a very primitive treatment of network activation compared to the sorts of models of memory in psychology and the network theories in artificial intelligence. From the evaluations with the architects of the CAD module we know that the representation is actually rather shallow leading to tutorials which tend to be rather boring for students to engage in. It is hard within the current system to provide the sort of strong lead or "story-line" which hand-crafted CAL can produce. The results often appear to students to be somewhat disjoint fragments of a multi-media glossary of related ideas. And finally, it is still quite hard for users to ask the right questions of the system as they often don't know what it is exactly they don't understand.

References

Nicolson, R.I. and Scott, P.J. (1986). Computers in Education: the software production problem. *British Journal of Educational Technology*, 17(1), 26-35.

Novak, J.D. (1983). Concept-Based Learning. In Boulding, K.E. and Senesh, L. (eds.) (1983). *The Optimum Utilization of Knowledge*, Westview Press, Colorado, pp. 100-113.

Scott, P.J. and Lawson, B.R. (1990). Generating Tutorial & HyperCard Assistance in the CAAD-Tutor. Paper presented to the Seventh International Conference on Technology and Education, Brussels, Belgium.

Scott, P.J., Lawson, B.R., and Ryu, J. (1991). An Intelligent Tutorial System for Computer Aided Architectural Design. Proceedings of the First International Conference on Artificial Intelligence in Design, Edinburgh.

Scott, P.J. (1993) Generating a HyperMedia tutorial from a Concept Network. CAL into the Mainstream: Proceedings of CAL-93. Hammond, N. & Trapp, A. CTI Centre for Psychology:York, UK. 167-8

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems.* Morgan Kaufmann, Los Altos, California.

Acknowledgements

This work was produced by a team including Dr J. Ryu, Prof. B. R. Lawson at Sheffield University.

Combining Associative Memory and Multimedia Technology for Training Patternmakers

MICHAL S. SOCLOF and YOH-HAN PAO
Department of Electrical Engineering and Applied Physics
Case Western Reserve University, Cleveland, Ohio 44106, U.S.A.

Abstract: We report on ongoing research in the design of a training system that combines the use of computer-based associative memory and multimedia functionality. The goal of the system is to provide means for helping the student learn the facts and fundamentals of the topic area, as well as to provide support in his practice of the art. The system facilitates associative recall of similar designs encountered in the past and supports troubleshooting and preemptive troubleshooting. Students are taught an important basic skill: the ability to describe designs in terms of meaningful geometric and descriptive features, and to describe the imperfections in terms of standard defect classes and causal features. These linguistically expressed features serve as the interface between the multimedia training system and the student/practitioner.

Introduction

In this paper, we report on ongoing research in the combined use of computer-based associative memory and multimedia functionality, in the teaching of complex tasks. We find that of critical importance is the matter of teaching interpretive and judgemental skills. With those abilities in hand, the learner can access computer-based associative memories as well as his own memory to recall previous experiences, to enhance efficiency and to anticipate problems. Multimedia capabilities and appropriately structured computer associative memories are useful not only for the teaching of such skills but also in the practice of the art.

In the manufacturing industry, many complex tasks are currently taught through an apprenticeship program. Through examples and instructions, the expert craftsman relays to his charge the expertise gained through the years. It is this experience that sets apart the master craftsman from the novice, allowing the master craftsman to draw upon past mistakes and achievements to produce new products correctly and efficiently. The experience and expertise gained through years of practice is invaluable to the industry and can be lost if not properly captured and transferred. Our work addresses the question of how this information might be captured and transferred with the help of computer-based associative memory and multimedia technology.

Our goal is to help the student develop the judgment, skills, and experience required to become an expert in a topic. We teach not only the facts but, more importantly, the ability to form interpretations and make useful associations. The approach is to augment the apprenticeship program with the use of a computer-based memory system that gives the student freedom to explore the use of the associations built into the memory. Furthermore, we add the use of different forms of media for interaction and presentation, in order to provide a descriptive mechanism that may help shorten the learning cycle and teach the necessary interpretive skills. The interaction between the associative memory and the multimedia techniques provides a powerful and flexible tool to help the user build his own skills and concepts.

In the next section, we describe the topic that we selected for exploration purposes, namely patternmaking for the metalcasting industry. We then describe our early achievements for teaching facts and fundamentals. In the final section, we discuss ongoing research and design of the overall system, including associative memory, troubleshooting and the teaching of interpretive skills.

Patternmaking and Metalcasting

We chose to concentrate on the metalcasting industry, specifically in the area of patternmaking. A metal casting is an object that is produced by pouring molten metal into a mold cavity and then allowing the metal to harden. The process of metalcasting involves several steps and is staffed by highly trained individuals.

After the design engineer creates the design and sets the specifications for the part to be cast, it is the job of the patternmaker to translate the engineer's blueprint to a solid three-dimensional form called the pattern. In part, the pattern is a duplicate, in shape, of the final casting. However, in addition, incorporated in the pattern is a system of channels, called the gating system. The gating system is necessary to properly feed the molten metal to the cavity. The geometry of the part may also need to be adjusted in order to be able to properly remove the pattern from the mold, while leaving the mold intact. This means for example, that draft is required. Draft is the taper on the vertical sides of the pattern.

The pattern can be made from different types of materials: wood, plastic, or metal. The mold can also be made in different ways using a variety of materials. A common process is the green (meaning moist) sand molding process. The mold must be made while the sand mixture is still damp.

The simplest way to make a pattern is to make it in one piece. This is not always possible because certain shapes cannot be withdrawn from the mold while leaving the sand intact. For example, there are many castings that have hollow parts or internal protuberances. Patterns that provide for these features cannot be withdrawn in one piece. Cores and loose pieces, however, allow these types of castings to be poured. The core limits the flow of metal in the mold. The core is put into the mold right before the molten metal is poured. Keying devices, called coreprints, are used to help hold the core in place, in the mold, during pouring.

A mold is made in a hollow rectangular box, called a flask. The mold is divided into two parts: the cope, which is the top half, and the drag, which is the bottom half. The line on the casting corresponding to the separation between the cope and the drag is called the parting line.

The issue of draft is directly related to the parting line. The placement of the parting line impacts the amount of draft required as well as which surfaces should be drafted. Another concern associated with both parting line and core design is the tradeoff between designing the pattern with cores versus using the technique called "cope-down." Under certain circumstances, the patternmaker can choose whether to "cope-down," that is to design the cope to be extended into the cavity, or whether to have a flat cope with a core placed in the cavity. This decision is up to the patternmaker since usually, the quality of the pattern and casting will be equivalent regardless of the decision. This issue is very difficult to explain with text alone and is an example of why multimedia technology is helpful.

The concepts described in this section are illustrated in Figure 1. In this figure, we see a hollow cylindrical part to be molded and the drag of the mold, with cores in place. The mold is for fabricating two castings at the same time and shows two cavities side by side. The cope is not shown.

Early Achievements

We designed a system with the intent to teach the facts and fundamentals of a specific subtask of the general practice of patternmaking as well as to illustrate how the associative recall of similar previous designs could be used in the training system. In designing the prototype system, we want to lay the groundwork to visually display different pattern designs in order to explain the process of patternmaking. This prototype would also let us design the framework for a more complete training system. Furthermore, it would establish the structure that would allow us to show and discuss different parts, view different designs, as well as provide the tools to teach difficult concepts.

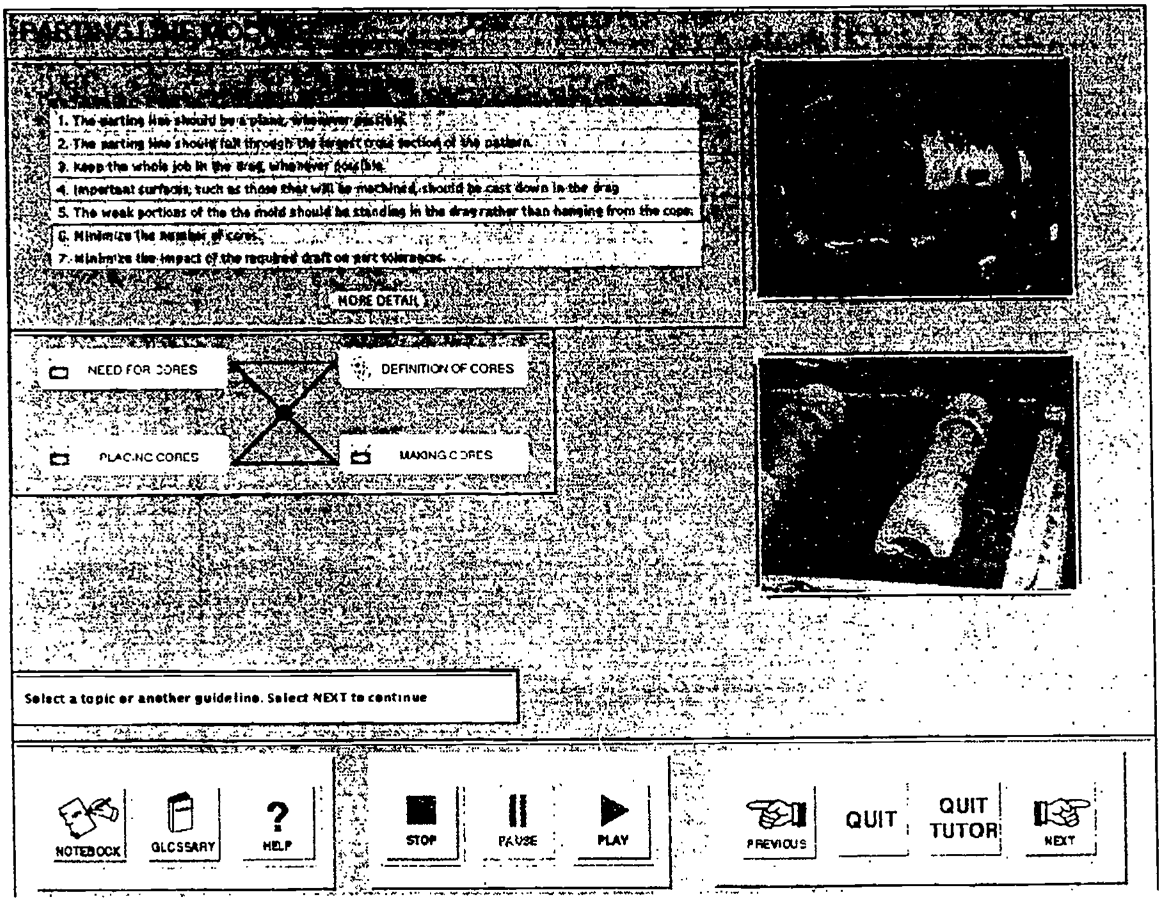


Figure 1: Prototype System Screen for Guidelines to Location of Parting Line

For the prototype, we chose to concentrate on an important aspect of patternmaking: the location of the parting line. As mentioned previously, the parting line is the line on a casting that corresponds to the separation between the cope and the drag. The placement of the parting line is one of the first and most important decisions that a patternmaker must make when designing a pattern. It is important because it will affect other design decisions for that pattern. For example, the amount of draft required as well as which sides of the pattern must have draft is directly affected by the parting line location. Furthermore, the decision whether to "cope-down" or make cores is also bound to the location of the parting line. There is no body of general hard-and-fast rules that can be used to decide where to place the parting line. Each decision must be made individually, based on the geometry of the part and other considerations. What complicates the matter even further is that there may be several correct parting line decisions for each design. The patternmaker chooses the placement of the parting line based on his knowledge, past experiences, and personal preferences.

In order to teach parting line decisions, we have to present the information at a conceptual level. Furthermore, we want to be able to provide knowledge both in the form of meaningful guidelines and related experiences. The guidelines will present the fundamentals and the experiences will provide the means for the novice to gain expertise based on the successes and failures of other patternmakers.

Upon entering the system, the student is presented with a brief definition of a parting line using a live video image, a still and an audio description. Seven short guidelines on the placement of the parting line are then presented. The student can request more information on any of the guidelines by selecting that specific guideline. Based on his choice, a detailed explanation of the guideline and several examples are presented to the student. These examples may be shown in the form of graphical images with audio explanations or videos.

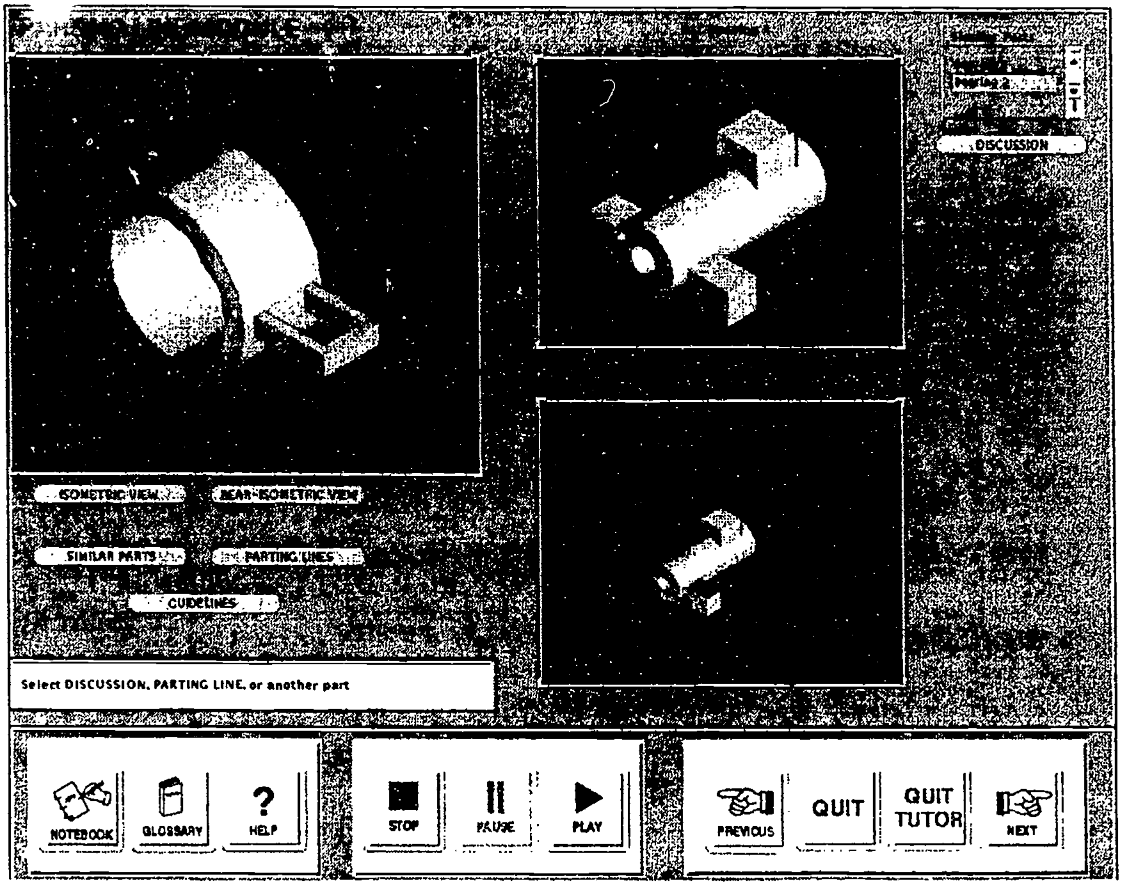


Figure 2: Retrieval of Similar Previous Part and Its Solution

Whenever any piece of information or instruction is presented, the student may choose to have it repeated as many times as he would like. Furthermore, additional topics that are related to his query and the interconnection between them are shown. The student can then choose to study any of the associated topics. When ready, he can proceed. Figure 1 shows a prototype system screen during this interaction.

Next, the student is shown a new part and different choices of parting lines for that design. He can view the part from different perspectives. If he is ready to make a selection, he can choose the parting line that he thinks is best. He will then get a discussion about the part and the solution, including which guidelines were followed, how the other solutions compare, and references to associated designs. If the student is not ready to make a decision, he can explore the related experiences of other patternmakers that are stored in the memory. If he chooses that option, he will then have a choice of several associated parts. In viewing each associated part and the correct solution, the student can learn from past experiences of other patternmakers and learn why the parting line decision was made for each case. The student may then be able to infer the solution for the part that he is currently studying. Figure 2 shows a prototype system screen during this interaction. In this figure, for example, the hollow cylindrical section in the associated design would teach the student how the parting line placement affects the need and design for a core, which could then be applied to the hollow cylindrical section of the part in question. When the student feels comfortable with making a decision for the part in question, he can return and make the choice. At any time, the student can explore the information contained in the memory and the associations presented to him. He can study the topic of interest, viewing more examples, videos and instructions.

Ongoing Research and Design

After using the system for training in learning the facts and fundamentals of the patternmaking task, we would like the student to be able to use the system and its multimedia capabilities for guidance during the practice of the art. Using the system as a guide would enable the practicing patternmaker to effectively apply the associated experiences of previous patternmakers as well as allow him to troubleshoot defective parts and patterns. Given the complex nature of these goals, we would also like to incorporate into the training system the lessons necessary to teach the trainee how to effectively use the system as a guide.

In order for the patternmaker trainee to be able to effectively use the associative memory, it must be structured so that the trainee can interact with the memory in a natural and comfortable manner. The associative memory, described generally in Pao (1992), is structured so that designs with similar features, such as geometry, are associated together. It is not enough in this context, however, to just use geometric descriptions. In order for the memory to be useful for the patternmaker trainee, the part and the pattern should also be described in terms of part and pattern descriptive features that are appropriate for that field, for example, sharp corners or thin holes. These features are more natural and relevant for the student to use in describing parts and patterns for both design and troubleshooting purposes. Geometric features such as large section, long thin rod, flat plate and so on, in addition to the descriptive features, are useful for remembering parts on the basis of geometric designs. The student can be taught the skill of describing designs in terms of meaningful descriptive features by having the system show him several different designs, using various types of media, with their corresponding appropriate design features. Through practice, he can learn this interpretive skill.

The associative memory can also be very useful for troubleshooting defects and potential defects. Effective troubleshooting requires an additional skill that should be taught by the system. This skill is the ability to interpret or describe an observation of a problem on a part or pattern in terms of a named defect category commonly used in the field of patternmaking. In order to teach this skill, the system would show the student, through videos, pictures and graphics, examples of defective designs and patterns and then describe which features characterize certain defects and the actual name of each defect. This interpretive skill is the first step in a procedure to identify and correct defects, as shown in Figure 3. After the defect has been named, there are several contexts in which it can occur. For example, the defect may be due to inappropriate design or due to faulty pattern equipment. At present, we are concentrating on the pattern design context. For a given context, there are causal features that could have led to or *caused* these defects. At this step, another interpretive skill comes into play. The patternmaker trainee must now see if any of these causal features correspond to the pattern descriptive features for the given pattern. That is, he checks if his pattern contains any problematic features in the current context. If so, the student can then look up in the memory the cure associated with the problem. If not, he can check another context and its associated causal features. The student can also explore how the observed defect may be due to one of several different causes in different contexts. Using the examples stored in the memory, the student can ask for examples to learn the subtle signs that would be helpful in differentiating among several potential causes.

The memory and methodology used for troubleshooting can also be employed in identifying potential defects for a certain pattern design or a given part. This can be achieved by using the pattern descriptive features or the part descriptive features that the patternmaker trainee enters. The trainee, with the assistance of the system, can then check in the memory to see if a part with these features could correspond to a defect. The cure could then be incorporated into the design.

The methodology used to teach the design, troubleshooting and associative skills discussed is important in order for the student to truly benefit from the system. Several guidelines are used throughout the system. Specifically, during a learning session, the student is provided with a recommended learning path. He can, however, veer off and explore any area that sparks his interest, or that he would like to study further. He can continue in this manner until he is ready to proceed with the lesson. Therefore, the student is able to get help and information when he needs and wants. This type of system provides the student with a non-threatening

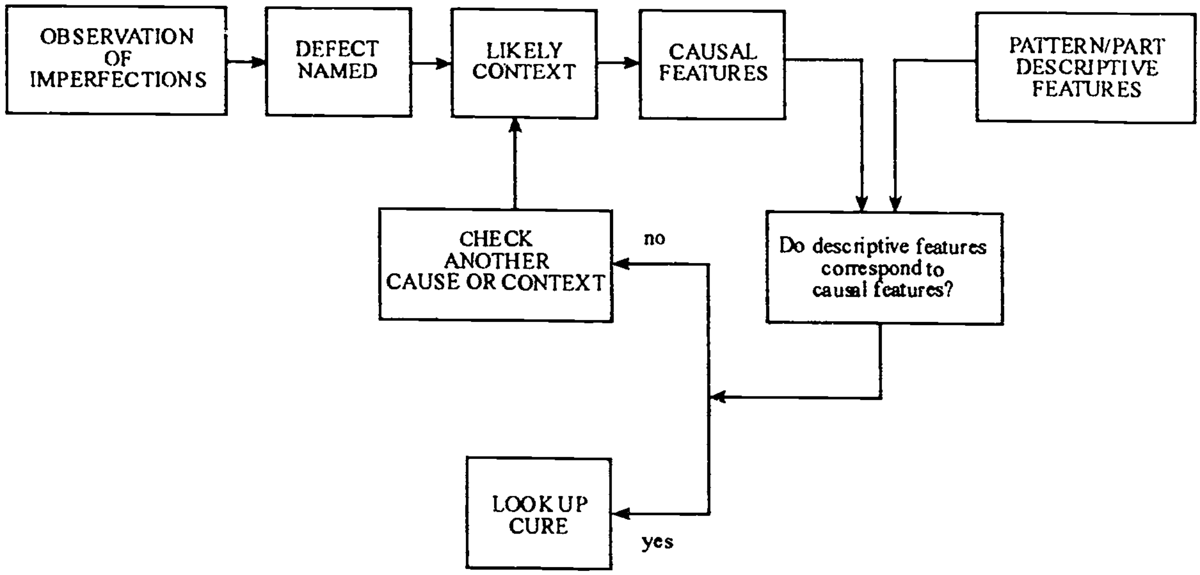


Figure 3: Procedure for Identifying and Correcting Defects

environment. He can continue to ask questions and explore areas of interest without frustration or fear of embarrassment. The learner can test himself to see if his interpretive skills are developing well or if he needs more examples and practice in a particular context or for a particular defect. He can test his skills by asking the system to provide him with more cases in a particular area.

The various types of media are important tools used in the design and implementation of this system. Given that patternmaking is both an art and a craft, a hands-on skill, it cannot be completely taught simply by classroom lecture instruction. The student must see different examples and be able to graphically try different techniques. Patternmaking is also a complex task. It is, therefore, important that the training and relaying of experience is presented appropriately for each given student. Various students may assimilate information differently. One person may find verbal instruction more useful, whereas another, may prefer a picture or diagram. Different techniques or combinations of techniques may be helpful in learning new material or reviewing information. Furthermore, certain information may be better transmitted in one form of media versus another. With multimedia technology, video, audio, text, graphics and pictures can be used to help a person learn new concepts or improve his knowledge in areas he has already studied. In this research, multimedia is used to ensure that all sensory channels available are being utilized, in text, in sound, in pictures and eventually, when possible, by feel. In order to be more effective, multiple forms of media are also sometimes used together.

These procedures and techniques illustrate the advantage and power of using the associative memory as well as the multimedia technology as tools to train and guide patternmakers. The memory and the different forms of media let the student benefit from the experience of seeing many different designs, patterns and problems without having to perform the design process himself. The trainee can benefit from the experience and expertise of many master craftsmen, both from their mistakes and their solutions.

Summarizing Remarks

We have carried out a conceptual design of a new type of multimedia supported, self-paced training system and have implemented our design in several aspects of the system. The goals of the system are to train a potential patternmaker in patternmaking and to provide guidance and support during the practice of the art. After training in the facts and fundamentals, the system can be used to support both the recall of associated designs encountered previously and troubleshooting. The student is also taught to describe a design in terms of meaningful descriptive features and to describe an observation of a problem in terms of a named defect category.

It is our belief that the storage of experiences and expertise of many trained individuals, when augmented with multimedia techniques, can provide a powerful tool and effective means of capturing and transferring the knowledge required to perform a complex task. Our approach entails storing the knowledge and experiences in an associative memory that gives the student freedom to explore the memory structure. Furthermore, given the difficulty of teaching patternmaking with text alone, we need to employ different forms of media. These include video recordings, audio instructions and explanations, graphical images and illustrations, as well as text.

In this paper we present a brief report on ongoing research. Given that we had to implement, at a very low level, many of the multimedia functionalities required by the system design, we look forward to improvements and innovations in multimedia technology. Also, we continue to explore and develop the associative memory structure. We feel that this prototype laid the foundation and framework necessary to continue and build on our research.

Acknowledgments

We acknowledge the considerable help we received from the staff of the American Foundrymen's Society and the Canadian Tooling Manufacturers' Association. Of the many helpful and instructive publications made available to us, we list three in the bibliography as suggestions for further reading and for reference. We also thank Stephen Gregory and Benny Carreon of Kelly Air Force Base for their assistance and guidance in this project. Alok Mathur of the University contributed significantly in the implementation of the prototype system. We also thank Ron Cass of AI Ware for his continued interest and never-failing readiness to provide helpful technical assistance, including providing example parts. Part of the funding for this project was provided by the Cleveland Advanced Manufacturing Program.

Bibliography

Course Manual for Patternmaker Course (1988-1993). Cambridge, Ontario, Canada: Canadian Tooling Manufacturers' Association.

Hamilton, E. (1990). *Patternmaking Guide, Second Edition.* Des Plaines, Illinois: American Foundrymen's Society, Inc.

The Metalcasting Process Video (1986). Des Plaines, Illinois: American Foundrymen's Society, Cast Metals Institute, Inc.

Pao, Y.H., Komeyli, K., Shei, D., LeClair, S., & Winn, A. (1992). The Episodal Associative Memory: Managing manufacturing information on the basis of similarity and associativity. Special issue of the *Journal of Intelligent Manufacturing (JIM 1993)* on Neural Networks in Manufacturing, 4, 23-32.

Effective Use of Video in Interactive Modules

S. J. SUPRISE
Enzian Technology, Inc.
3251 Progress Drive, Suite A, Orlando, FL 32826, USA

N. L. MITCHELL
Director, Instructional Systems
Rollins College, 1000 Holt Avenue - 2647, Winter Park, FL 32789, USA

Abstract: Although interactive video instruction offers a powerful educational tool with which to provide students practical and engaging experiences in the classroom, its effectiveness depends upon its pedagogical design and strategic instructional use. In order to utilize the established advantages of interactive video instruction, instructional designers must apply what we know about how people learn to the development of this instructional medium. This article identifies common themes in learning theory that have practical design implications for the instructional designer of interactive video modules. The following "guidelines," extracted from these pertinent theories, are suggested to increase the effectiveness and appropriate use of IV instruction: 1) prepare the learner, 2) attract and direct the learner's attention, 3) guide the learner through successive steps of complexity, 4) present the material repeatedly in a variety of contexts, 5) provide a vehicle for practice with immediate feedback, and 6) make connections between new information and old information, showing how it fits into the "big picture."

In a generation of learners raised with television, music videos, and video games, it is no wonder that video has been hailed as a remedy to education's weaknesses. Many instructional developers boast of innovative applications for the myriad of new video "toys" available to media developers, and empirical studies validate the claim that interactive video instruction is effective. But because technology has made it so easy to incorporate video into educational materials, it is often incorporated haphazardly or misused. As with the development of all interactive educational programs, the integration of media must be done with pedagogical purpose and strategic instructional design. To facilitate the transfer of information and to provide an effective learning environment, we must adhere to sound instructional design principles, and use an appropriate medium to fit the instructional objectives, not mold the instruction to fit the tool.

Empirical Studies Discuss the Use of Interactive Video Instruction

Recent studies indicate that interactive video is generally an effective and efficient instructional medium. For example, interactive video instruction has been shown to be 30 percent or more efficient than lecture, also yielding greater retention rates than typical group instruction (Brandt, 1990). Dowding's literature review (1991) revealed that interactive video instruction led to a 40 percent reduction in training time with a 40 percent reduction in error rate; other studies report figures as high as 50-70 percent. Smith and Lehman's (1988) review of many interactive video programs yielded similar results, substantiating the claim that video training programs are efficient and generally as effective as traditional training methods.

Additionally. researchers report many other benefits of this media. Interactive video instruction generally holds students' attention as much as 34 minutes longer than "traditional" lecture methods (Petty & Rosen, 1987). Students also may access interactive video programs independently, progress through instruction at their own pace, and repeat segments or the entire program as desired (Garrett & Savage, 1990). By creating an environment where students are active participants in the learning process, they ultimately become accountable for their own learning (Nelson & Watson, 1991).

However, some suggest that video itself has no more impact on learning than any other media. Clark (1983) believes that "media are delivery vehicles for instruction and do not directly influence learning" (p. 453). Smith and Lehman (1988) state "that it may be the design of effective materials rather than the medium itself that produces the [research] effect" (p. 29). We believe *that it is the design of video materials and the way in which they are used in instruction that actually makes them effective, rather than an inherent quality of the medium itself.* Although research studies reveal the apparent effectiveness of interactive video, more research is needed to determine *how* it can be effectively used and *for which* learners and learning tasks it is optimal. As Tannenbaùm and Yukl (1992) state, "Empirical research to determine how different features of the high technology methods facilitate training has lagged far behind development of the technology itself" (p. 433).

The Appropriate Use of Interactive Video Instruction

It seems clear that video alone cannot solve all of education's problems, especially when it is used inappropriately, and in fact, it is often misused. Video clips that are nothing more than a view of a narrator talking to the camera, merely a "talking head", have little educational value and the instruction could be accomplished with lecture or audiotape. Yoshii, Milne, & Bork (1992) found that presenting long video segments of factual information to students before requiring any interaction tends to leave them uncertain about where to focus their attention and about which features were the most salient (p. 4). This is an inherent problem in the lecture method that interactive computer programs were meant to alleviate. As Brandt (1990) points out, "The danger in the continued use of weak designs is that someday, if interactive video becomes prevalent, it may be accused of the same shortcomings that school systems are being accused of today" (p. 10).

Another common misuse of interactive video is the use of an inappropriate level of interactivity. Interactivity means that the user actively participates in the learning situation and has at least some amount of control of the information presented. The learner's responses are usually used to further tailor the instruction to the learner's needs or to provide feedback by showing the consequences of one's choices (Petty & Rosen, 1987).

Video may be integrated into computer-based instructional programs at various levels of interactivity, each with its own advantages. Some instructional tasks require higher degrees of interactivity than others, just as some learners require more participation and activity in the learning process than others. At the lowest level, interactivity is typically limited to stopping or restarting a video segment. This level is all that is necessary when using video to place a concept into a relevant context, or to help students visualize concepts which are difficult to explain verbally. For example, short clips of scientific phenomena and procedures are especially effective for illustrating cause-and-effect relationships that would otherwise require laboratory demonstrations. Video segments are also valuable in a social work curriculum where students can view interaction in a real therapy session, allowing them to reflect on nonverbal behavior, proper questioning techniques, and group leadership skills without the pressure of guiding the actual sessions.

A second degree of interactivity gives the learner full control of the video presentation, with the ability to stop and start the video, rewind to a specific segment, or pause on a specific frame. This level of interactivity is useful in several contexts. In a classroom presentation, the teacher may first present a video segment to prepare the learner and to create the appropriate context for the topic. Then the teacher may pause the video, repeat certain segments or frames, and emphasize or explain critical features to which the learner may not have attend. In an interactive video program designed as an out-of-class exercise, students can repeat specific segments that were not clear when viewed initially, explore the segment again while attending to details, or merely pause and reflect on the new material before proceeding.

The highest interactivity level allows the learner to respond to video, altering what is seen and heard. Artificial intelligence incorporated into a computer program branches the learner to a specific video segment based on his or her response. Such highly interactive strategies also allow programs to collect information about student progress and provide feedback accordingly. This facilitates guided exploration, practice with immediate feedback, and opportunities for students to meaningfully apply what was learned in a non-threatening environment.

For example, in the Education Department of the University of Michigan, Professor Carl Berger created an "interactive fiction" project called Klepto (Staff, 1992, p.15). This videodisc is composed of a series of scenes in which a hypothetical classroom incident--in one case, the discovery of some missing classroom equipment--provides the viewer with a series of choices about how to handle the problem. After seeing a video segment, the viewer is required to make a series of decisions about how to respond to the incident. Through program branching, the viewer can observe various outcomes of their choices; "Each choice will lead a participant to specific consequences. The goal of the program is not to suggest a *right* way to handle a specific classroom problem, but to serve as a catalyst for class discussion and exploration of consequences of different responses" (p. 15).

Some other appropriate uses of interactive video instruction have been suggested by Dowding (1991). He maintains that interactive video instruction can be effectively used to 1) simulate operational consoles, in which students can practice actual operations on simulated equipment; 2) teach concepts and skills when immediate, objective, or visual feedback is required; and 3) teach knowledge or skills that could be enhanced through the use of visualization. For these uses, interactive video "can be used in the classroom to describe basic system characteristics, capabilities, and limitations; to show background and operational theories and principles; and to demonstrate complex relationships" (p. 308).

Interactive video instruction is particularly effective for training motor skills, procedures, or processes. Dowding (1991) states, for example, that interactive video is "extremely effective in providing instruction and performance in advanced operations and tactical decision-making through simulation of tactical situations" (p. 308). For these tasks, interactive video instruction can utilize behavioral modeling techniques to teach procedures or skills. The required skill or procedure can be displayed through a video presentation, an "expert" can demonstrate the correct performance while clearly delineating each step, and then students can be required to practice the procedure through simulation of the operation (p. 308).

The ultimate purpose of education and training is to enable students to use the skills and knowledge gained in the classroom to accomplish meaningful goals in the real world. Interactive video instruction that closely simulates that real-world environment can prepare students for those experiences. Interactive videodisc instruction can allow students to practice performing operations or to learn processes that are normally difficult or impossible to safely execute in the real world until the student has gained considerable skill and experience. For example, interactive video incorporating real footage is often used to train pilots to land a plane under emergency situations. Or, it is used to train students in the classroom to perform tasks that require expensive, easily damaged equipment or the execution of potentially hazardous procedures. In an educational environment, this type of interactivity permits students to freely investigate phenomena such as chemical reactions, in significantly less time than required for traditional laboratory exploration (Hoffer, et al., 1992).

The Incorporation of Instructional Design Principles

Consequently, for video to truly be a useful educational tool, it must be used for those educational tasks for which it is best suited and incorporated a level of interactivity that facilitates the maximum amount of learning possible. However, as Smith and Lehman (1988) warn, in order to use interactive video to create effective educational products, sound instructional design principles must be also be used.

In a previous article (Mitchell, Suprise, & Ray, 1993), we outlined several teaching principles that apply to the development of instructional materials. These important guidelines are based on information from many disciplines, such as Learning Theory, Information Processing and Problem Solving, Perception, Graphic Design, Cartography, and Instructional Design. Although often articulated with different terminology, there appear to be some teaching principles common to these perspectives: 1) prepare the learner, 2) attract and direct the learner's attention, 3) guide the learner through successive steps of complexity, 4) present the material repeatedly in a variety of contexts, 5) provide a vehicle for practice with immediate feedback, and 6) make connections between new information and old information, showing how it fits into the "big picture." These guidelines help ensure that the learner's attention is focused on the appropriate information, that the concept is presented in different context to promote generalization, that the learner has an opportunity to apply and interact with the information, and that this process will be mediated by guidance and frequent feedback. Thus, these six principles represent a condensation of the learning theory literature articulated as "design tips" backed by decades of empirical research and practical use. These principles may be particularly helpful when designing interactive video instruction.

Prepare the learner

Video can be used, for example, to prepare the learner by placing new information into a context that is relevant to students. It is motivating for students to see a relationship between what is presented in class and what occurs in their own lives. For example, by beginning a lesson on sleep physiology with a short video clip that relates sleep deprivation, a common problem for students, to alcohol use can make the study of this topic more interesting and engaging for students.

Attract and direct the learner's attention

Video also seems to naturally attract students' attention, but as we have previously discussed, with a minimal level of interactivity, it can also serve as a medium to help distinguish critical features of a new concept. Often, what appears obvious to the educator, may not be so obvious to students. Showing an appropriate video clip, and then returning to segments and discussing the salient features that are depicted can help students sort through and focus on the most important points.

Guide the learner through successive steps of complexity

Video can be used to guide learners through successive steps of complexity, reinforcing accomplishments as they proceed. As compared to the mere linear presentation of content, the incorporation of video allows the learner to progress from introductory clips that create a context for new concepts, to video segments that show applications of the concepts to real world problems, to interactive exercises that allow students to interact with video to practice what they have learned. In the sleep deprivation example cited above, video is used to guide the learner through deeper levels of understanding by first showing several circumstances under which sleep deprivation is studied, methods used to measure brain wave components during sleep, and finally still motion video frames depicting wave components present in sleep patterns. Students are encouraged to visualize more and more complex applications of these ideas and are given an opportunity to "measure" brain waves.

Present material repeatedly in a variety of contexts

Video can also facilitate learning through the incorporation of successive clips that place new content in varied contexts. Digitized video now allows designers to incorporate short clips from many sources into one seamless instructional piece. These "repeated presentations" have significant effects on cognitive development (Hoffer et al., 1992). Research has shown that when successive examples of concepts are separated in time, they are less likely to be active in working memory at once (Anderson, 1981, cited in Hoffer et al., 1992, p. 10). When students can observe effects in only a few minutes via interactive video, they are more likely to recognize patterns in the results unlikely to be seen in the fragmented view of the typical classroom or laboratory environment. Also, repeated presentations of concepts in varied contexts facilitates generalization.

Provide a vehicle for practice with immediate feedback

Most importantly, students need an opportunity to apply this new information in a meaningful way, not just as an isolated homework assignment, but through a variety of strategies that also provide immediate feedback for their efforts. A video program created by Dr. Michael Valentine (Knapp-Minick, Gottron, & Loven, 1991) called "Talk To Me Teacher" represents an excellent example of video used to provide hands-on practice with learned material. In this example, video is used by students to practice effective communication skills for classroom discipline. A video segment is presented showing a typical disruptive classroom incident. As the teacher in the video attempts to diffuse disruptive behavior, the student records any instances of unclear communication displayed by the teacher. Then the student is given a choice of statements to use to intervene and immediately sees the effects of this choice; either the disruptive behavior ceases or escalates. Students can practice classroom intervention and communication skills in a non-threatening environment, repeat the exercise, and receive immediate feedback regarding their progress.

Make connections between new information and old information

Finally, implementing these instructional principles helps to ensure that students make connections between what they have learned in the past to new information, helping them to formulate "the big picture" and to show how each element fits. These interconnections enable learners to combine ideas, infer, extrapolate, or otherwise reason from them and thus formulate concepts "beyond the information given" (Bruner, 1973, cited in Hoffer et al., 1992). Video is an especially effective tool for helping students make these connections because it can present real life applications of concepts that could not otherwise be created in the classroom. As Dede (1987) concludes, "A delivery system of visual media capable of supporting learner interactivity while at the same time facilitating interconnectivity of images and symbols has the potential to become an extremely powerful education tool" (cited in Hoffer et al., 1992).

In conclusion, when approaching an instructional need, it is important to carefully consider the different instructional media available to determine how that need could best be satisfied. In some cases, interactive video instruction will satisfy that need; in some cases, it will not. Yet, video designers and computer developers often apply this emerging and promising technology to many instructional problems based on its known merit and unquestionable power. The result is a very expensive interactive video module that yields "no significant difference" or that results in learning that is equivalent to what could have been accomplished through simple stand-up classroom instruction or laboratory work.

Therefore, video can be an effective educational tool if it is used for those applications for which it is best suited; to introduce material, to help learners visualize the key points of a concept, to relate them to other concepts, and to help students see real world applications of ideas explored in the classroom. Furthermore, video must be incorporated into the lesson with strategic planning according to accepted principles of instructional design. It can be used to promote understanding by attracting and directing students' attention to critical features of the material, by placing it in a context that is familiar to them and which gradually increases in complexity, and by allowing an opportunity for students to apply this information to promote connections to what they already know. If these principles are carefully applied to the design of video in interactive modules, educators can begin to capture its powerful educational effects.

Referenees

Brandt, R. (1990). Interactive means tutoring not teaching. *Journal of Interactive Instruction Development*, 2(4), 8-10.

Clark, R. E. (1983) Reconsidering research on learning from media. *Review of Educational Research, 53*, 445-460.

Dowding, T. J. (1991). Interactive video... The future of training technology. *International Journal of Instructional Media*, *18*(4), 305-312.

Garrett, T. J., & Savage, D. G. (1990). A comparative study of the teaching effectiveness of a slide-lecture and microcomputer-videodisc program. *Journal of Interactive Instruction Development*, 2(4), 26-27.

Hoffer, T., Radke, J., & Lord, R. (1992). Qualitative/quantitative study of the effectiveness of computer-assisted interactive video instruction: The periodic table of elements. *Journal of Computer in Mathematics and Science Teaching*, 11, 3-12.

Knapp-Minick, B., Gottron, C., & Loven, J. (1991). Talk to me teacher... implications for interactive video as a teacher-training tool. *Journal of Interactive Instruction Development*, 4(2), 19-24.

Mitchell, N. L., Suprise, S. J., & Ray, R. D. (1993). *Bells and whistles*. Proceedings of the Orlando Multimedia '93 conference sponsored by the Society for Applied Learning Technology, 47-50.

Nelson, C. S., & Watson, J. A. (1991). Multimedia and repurposed interactive videodisc courseware: The means to teacher authored content-rich learning materials. *Journal of Interactive Instruction Development*, 3(3), 10-15.

Petty, L. C., & Rosen, E. F. (1987). Computer-based interactive video systems. *Behavior and Research Methods, Instruments, & Computers*, 19(2), 160-166.

Smith, E. E., & Lehman, J. D. (1988). Interactive video: Implications of the literature for science education. *Journal of Computers in Mathematics and Science Teaching*, 8(1), 25-31.

Staff. (1992, April/May). Using interactive video in instruction at the University of Michigan. *Syllabus*, p. 15.

Tannenbaum, S. C., & Yukl, G. (1992). Training and development in work organizations. *Annual Review Psychology*, 43, 399-441.

Yoshii, R., Alastair, M., & Bork, A. (1992). Strategies for interaction: Programs with video for learning language. *Journal of Interactive Instructional Development*, 5(2), 3-9.

History, Hypermedia, and Criss-Crossed Conceptual Landscapes: Designing Hypermedia Applications to Support the Development of Historical Thinking

KAREN SWAN
State University of New York at Albany, Albany, NY 12222

Abstract: This paper is concerned with the design of hypermedia applications that support the development of historical thinking. It focuses on the design of a particular hypermedia application, *Set On Freedom*, and presents preliminary research, both qualitative and quantitative, on its use in a seventh and eighth grade classroom. Findings, which include an increased propensity to discover links between historical data and more empathetic understandings of historical events after using the application, suggest that hypermedia can indeed be designed to support historical thinking.

History is an ill-structured domain. Thinking about history entails more than simple familiarity with important facts and concepts; it involves being able to conceptualize historical events from multiple perspectives and to relate diverse historical data within these. Historical thinking, according to Booth (1984), is neither inductive nor deductive. It is rather critical thinking that is divergent, subjective, and focused on "human situations and the complex web of relationships embedded in them." (Langer, 1992, 3) Downey (1991, 1) defines historical thinking as the ability "to empathize with people of other times and cultures; to see relationships, including causal relationships, over time; to formulate concepts of historical time; and to distinguish between the past (everything that ever happened) and history (what we make of it)."

Teachers of history would like to elicit historical thinking among their students. They want their students to, in some sense, participate in the kinds of conversations and activities in which historians participate. Students, however, are not historians. They lack both the knowledge and the skill to "do history." The problem is how to encourage students to create their own meanings around historical events while providing them with sufficient historical information and enough structure to do so (Seixas, 1993).

That is where hypermedia comes in. Hypermedia has enormous potential for supporting the development of historical thinking because it is open, nonlinear, and heterogeneous in ways other media is not. Of particular interest in this regard is Spiro and Jehng's (1990, p. 170) notion of the "criss-crossed conceptual landscape," of critical understanding in ill-structured domains as evolving through the revisiting of similar materials from differing conceptual perspectives. Hypermedia, they argue, supports nonlinear, associative links between materials that make explicit this criss-crossing process. Hypermedia can thus physically represent information in ways that model the cognitive representations characteristic of critical thinking in ill-structured domains. Materials that model cognitive processes have been shown to aid in the development of similar patterns of thought (Salomon, 1981; Shapiro & Spoehr, 1992; Swan & Black, 1993).

Hypermedia has another advantage over traditional text-based materials for the development of historical thinking. Hypermedia supports the integration of a variety of media, video in particular. Video promotes affective understandings (Greenfield, 1984). The integration of video and text in hypermedia can thus encourage the utilization of differing mental representations to think in ways that are both empathetic and analytic.

This paper is concerned with the design of hypermedia to support the development of historical thinking Its purpose is to demonstrate that hypermedia materials can be created which foster students' critical thinking about history by focusing on the design of a particular hypermedia application, *Set On Freedom*. The first part of the paper discusses that design; the second part of the paper describes the results of preliminary classroom research.

Design of *Set On Freedom*

Set On Freedom is a hypermedia application that combines text, graphics, and video footage in a rich collection of images and information about the American civil rights movement It covers the period in history ranging from the Brown v. Board of Education Supreme Court decision ending legal segregation in public schools (1954), to the demolition of Resurrection City in Washington, DC (1968). It combines textual information describing the people, places, issues, and events of the period, video footage, still photographs, primary source documents, maps, and a timeline in a single screen, full color, *ToolBook* application. The application also contains a variety of structures for organizing and navigating this information in ways we hope will lead students to multiple views of the conceptual landscape (Spiro and Jehng, 1990) it describes.

Heller (1990) and others (See Hammond, 1989; Nelson & Palumbo, 1992) suggest that hypermedia systems be provided with navigational structures, such as indexes and maps, which continuously reorient users to their conceptual structures. Spoehr (1992) has found that differing organizations presented through such tools can lead to differences in users' conceptualizations of the information they contain. In designing *Set On Freedom*, our aim was to give users navigational tools which not only continuously reorient them to its conceptual structure, but which themselves offer multiple structural orientations to the materials.

Macro Organization

On a macro level, *Set On Freedom* organizes information about the civil rights movement from four different general perspectives -- *People*, *Places*, *Events*, and *Viewpoints* -- which we hope will encourage and facilitate students' development of deeper, more thematic perspectives of their own. Each perspective is instantiated as a submenu with a graphical interface that is accessible from anywhere in the application. The explicit structure of the application takes its users from the submenus to the screen(s) containing information on specific menu items and then back again to the submenus reorienting them to these structures. When a user chooses a particular item in a submenu, it turns gray to indicate that it has been visited, allowing the user to systematically explore a particular perspective.

The *People* section is directly accessed from an alphabetical listing that includes digitized photographs of the people named. It contains short biographies of seventy-four people who figured prominently in the civil rights struggle, their pictures, video footage, and writings by or about them. It is hoped that the design of this section will encourage students to make connections between people, to see the influence of events on ordinary people's lives, and, most especially, to see the influence that ordinary lives can have on history.

The *Places* section is directly accessed through a map of the United States. It provides geographical as well as historical information about twenty-eight cities and video footage of the important civil rights activities that took place in them. Screens in the *Places* section also include state maps which show the location of the city being discussed and the locations of other cities in the state that are included in the section. Clicking on the place markers of these other cities will take the user to the screens which discuss them. It is hoped that this spatial design will encourage students to make geographical connections between historical events, and to develop a sense of the influence place has in history.

The *Events* section of *Set On Freedom* describes forty-five major civil rights events in some detail. Each event is accessed from a scrolling timeline that, it is hoped, will encourage students to develop concepts of historical time and to see relationships between historical events. Clicking on a date on the timeline pops up a list of events that happened in that year. Clicking on the event takes the user to screens giving a narrative description and video footage and/or still photographs of the event. Many screens in the *Events* section are also linked to primary source documents.

The *Viewpoints* section is accessed either from a list of the twelve topics discussed, or from an alphabetical listing (similar to the *People* menu) of the eighteen people who discuss one or more of these topics. It adopts a present-day perspective to present the video-taped reflections of a variety of people. Discussants in this section address thematic issues and provide sometimes controversial commentary that we hope will start students thinking more critically about the civil rights movement. *Viewpoints* screens also include text describing the topic or giving short biographies of the people speaking, and frequently link to primary source documents.

Micro Organization

On a micro level, hotwords embedded in the text of most screens, linking icons located outside that text, and a topical index located outside the main sections of the application create a secondary, web-like structure through which users can criss-cross the conceptual landscape encompassed by *Set On Freedom*.

Hotwords embedded in the text of the application are shown in blue. Clicking on them will take users to other screens in that or differing sections of the application which examine the item from a different perspective and/or in greater detail. Besides the screens accessed through the sectional submenus, *Set On*

Freedom also contains screens which give expanded information on various organizations, movements, historical events, governmental actions, etc. which don't exactly fit within the given perspectives. These screens can only be accessed through the hotwords and index listings and the secondary web they help create

In addition to hotwords, many screens also have linking icons located outside their text on which the user can click for explicitly differing perspectives. Every screen in the *People* and *Places* sections, for example, has arrows that will take the user to screens describing the events in which they figured prominently Many of the *People*, *Places*, and *Events* screens have icons linking them to topics in the *Viewpoints* section. From every topic screen in the *Viewpoints* section, the user can choose to find out more about the people speaking and hear them speak on other topics. From every person screen in that section, the user can choose to find out more about a particular topic and hear what other people have to say about it.

Finally, a topical index, similar to those commonly found in printed books, is accessible from all screens in *Set On Freedom*. The index is arranged alphabetically by topics, with all the major references to each such topic in the entire application listed under its headings. Both these headings and the individual references are "hot" -- clicking on them will take users to the page or pages on which they can be found. From the topical index, an index of primary source documents is also available. The index of primary source documents provides access to the thirty-eight texts associated with *Set On Freedom*, and can be viewed by title, author, or date of publication

The hotwords, the linking icons, and the topical index provide three conceptually different ways that users can follow their own associative paths through *Set On Freedom* (Spiro and Jehng, 1990). The hotwords are embedded in its narrative and descriptive passages and are most like elaborations of, or digressions on, single ideas. The linking icons are separate from the texts, and the links they provide are named, more like cross-references to people, places, events, and/or reflections that are related to those being covered. The topical index, like its print analog, provides sets of entries collected under single headings but accessible as sets only through the index listings. It is hoped that by giving users multiple concrete representations of these differing kinds of associative connections we will encourage them to think relationally about history without leading them to believe there are "correct" interpretations of it.

Formal Structure

Set On Freedom also encourages a criss-crossing of its conceptual landscape on a formal level. All of the screens contain text describing the people, places, events, and topics involved. Almost all screens also contain video icons on which the user can click to view video footage and/or still photographs of a person, place, event, or reflection. Many screens contain primary source icons on which the user can click to view the text, or excerpts from the text of, thirty-eight important documents from the period covered by the application. All the screens in the *People* section contain still images of the people they describe. All of the screens in the *Places* section contain maps Thus, each individual screen in the application can be criss-crossed from the variety of perspectives afforded by a variety of media representations, including the important affective perspectives provided through video.

It has been well documented that video is a powerful tool for affective learning (Comstock, 1978; Howe, 1983, Greenfield, 1984). Video has been shown to change people's attitudes, values, and beliefs (Howe, Greenfield, 1984), to influence their behaviors (Comstock, 1978), and to affect the ways in which they perceive the world (Howe, 1983). Because it is a representational medium, video has the added value of giving its viewers a sense of time and place that cannot be gotten through any other medium. It is thus additionally supportive of the development of holistic and empathetic understandings of history.

Set On Freedom includes often dramatic, historical video footage that we hope will provide its users with a better sense of the people, places, and events of the civil rights movement, and help them to empathize with, and so understand, its concerns. The application also contains one half hour of the videotaped reflections of a variety of people that capture their feelings about the civil rights movement, and so, personalize and make the period covered by the application more accessible to students born a decade or more later.

In addition to video footage, *Set On Freedom* provides links to primary source documents (i.e. the Brown v. Board of Education Supreme Court decision; the Civil Rights Act of 1964; the Kerner Commission report; Malcolm X's "Message to the Grass Roots;" Ella Baker's "More Than Just a Hamburger;" Martin Luther King, Jr.'s "Letter From Birmingham City Jail;" the SNCC "Statement of Purpose"). Primary source documents are the raw materials with which historians work. It is hoped that by providing links to these resources within the structure afforded by the application, we can both give students a feel for the "doing of history" and put such resources in an understandable context.

Differing media use differing symbol systems to convey information (Salomon, 1981). Text is analytic and reflective; graphics are spatial and relational; video and still photographs are representational and affective. The integration of these differing sources of information in hypermedia encourages viewing a single topic from the variety of perspectives afforded by a variety of symbol systems. *Set On Freedom* was designed to bring together materials in differing media formats within a unified structural context, such that each could add to and complement the representations afforded by the others.

Preliminary Research

Methodology

In late May, 1993, we began investigating the use of *Set On Freedom* with students in a combined seventh and eighth grade classroom in a public school in a rural Vermont community. All students were white, generally middle class, and between twelve and fourteen years old. Groups of three and four students were asked to explore the application for a period of one hour and a half. Students' on-line sessions were videotaped.

All students were interviewed before and after using the application and both these interviews were also videotaped. At both interviews, students were asked to tell what they thought "civil rights" meant, what they knew about the American civil rights movement, and whether the American civil rights movement was meaningful to them. At both interviews, students were also given printed sheets naming twelve people, places, issues, and events and asked to identify items they recognized and draw links between items they thought were related. Two versions of the sheets were created so that students could be given differing versions before and after using the application. Half the students were given one version and half the other version at the initial interview (and visa versa at the final interview) to control for possible differences in students' knowledge of the differing items on the two versions. At the final interview, students were also asked to comment on their use of *Set On Freedom*.

Results

All students had very positive reactions to their use of *Set On Freedom*. In fact, students were so engaged in reading the texts, watching the video, and listening to the reflections of people who had direct experience of the civil rights movement that we were somewhat disappointed in their lack of verbalization for the videotaping. The tapes show engagement, but not the discourse we had hoped they would elicit. We hope that with further usage, and perhaps with task orientation, such conversations will evolve.

Another positive finding was that students had much clearer conceptions of the historical civil rights movement after using *Set On Freedom*. Before using it, the majority of students identified the American civil rights movement as a general movement to gain equal rights for all people In fact, in the initial interviews, women's rights were more frequently mentioned in this connection than were the rights of African-Americans After using *Set On Freedom*, the majority of students identified the American civil rights movement specifically as black Americans' struggle for equal treatment. In the final interviews, students were also more likely to name specific rights -- the right to vote, the right to use public facilities, etc -- in connection with this question, and students often associated their answers with visual images. This result suggests students gained a better sense of the historical civil rights movement through the video clips included in the application.

Perhaps more importantly, students more frequently stated that the civil rights movement was meaningful to them after using *Set On Freedom* than before using it. One boy, for example, initially said that because he was a white male living in Vermont, the civil rights movement was not important to him After using the application, he said that the civil rights movement was very meaningful to him because he would not want to live in a country where African Americans were treated so poorly. Many students had similar responses after using the application, and, interestingly, many students once again referred to specific images from *Set On Freedom* in these answers The result suggests that video can indeed be a catalyst to empathetic understanding of historical events, and clearly deserves further investigation

There were also hints that visual imagery might serve a kind of mnemonic, organizational function vis a vis the cueing and chunking of students' mental representations of complex materials When identifying items on the printed response sheets, students frequently first described a visual image from the application, and then gave a verbal explication of it. It seems possible that students were using these visual images as entrees to the information in their memories about the people, places, and events they were discussing. One is reminded here of the Greek art of memory theater, a technology of memory whereby pre-literate Greek orators would memorize long speeches by associating different parts of them with objects within the building in which they were to be delivered. As they delivered a speech, the orators would then use the objects as visual cues to call up the associated parts of the speech. Memory theater thus might be a useful analogy for hypermedia design, and visual images purposely employed as markers for complex ideas and/or sets of ideas.

Students' responses on the printed response sheets were recorded in terms of the number of items identified and the number of links recognized. They were analyzed by comparing means in both categories before and after using the application using single-sided t-tests. Table 1 gives the means and standard deviations of the number of items students identified on the response sheets before and after using *Set On Freedom* Table 2 gives the means and standard deviations of the number of connections students made between items before and after using *Set On Freedom*.

Table 1
Means and Standard Deviations of Items Selected Before and After Using Application

	Mean	SD
Before	4.89	2.05
After	6.05	1.96

Table 2
Means and Standard Deviations of Links Drawn Before and After Using Application

	Mean	SD
Before	1.00	1.33
After	2.74	1.99

The results show that students perceived significantly more linkages between people, places, issues, and events after using the application ($t = 5.362$; $p < .01$), even though they only identified slightly more items in the final interview ($t = 1.205$; $p > .10$). It is not surprising that students did not identify significantly more items after using *Set On Freedom*. The groups saw on average only 15 of the over 350 screens in the application. Indeed, the lack of significantly more items identified in the final interview makes the fact that students recognized a significantly greater number of links between the items more meaningful. If the students had identified both more items and more links, it would not have been clear whether their identification of more links resulted from a greater propensity to search out links or simply from the fact that they were familiar with more items. That students perceived more links without recognizing more items suggests that their use of the application encouraged them to make more connections between historical facts and concepts.

Discussion

Our preliminary research, then, suggests that hypermedia applications can be designed in ways that support students' development of historical thinking. In particular, the finding that students identified more links between items, but not more items, on the printed response sheets after using *Set On Freedom* points to the efficacy of a particular hypermedia structure for encouraging a particular kind of historical thinking, the ability to see relationships between historical events. (Downey, 1991; Langer, 1992). It seems reasonable to assume that students' use of the explicit, interactive links embedded in *Set On Freedom* prompted them to adopt similar, relational habits of thought (Salomon, 1981; Spiro and Jehng, 1990; Shapiro & Spoehr, 1991; Swan & Black, 1993). The preliminary research also suggests that video included in hypermedia may help students develop clearer and more empathic understandings of people, places, and events of other times and cultures (Downey, 1991; Langer, 1992). These results clearly deserve further investigation, as do several other issues addressed in the design of *Set On Freedom*, but not in the preliminary research.

One issue clearly not addressed by the preliminary research is the potential of hypermedia for helping students to develop rich understandings of historical periods through revisiting materials describing them from a variety of conceptual perspectives (Spiro & Jehng, 1990). Not only would such development require far more time than was available for this preliminary study, but, in our opinion, could only take place when mediated by a good teacher. A similar observation can be made concerning the development of rich conversations about history (Langer, 19992; Seixas, 1993). Our preliminary research suggests that students using *Set On Freedom* were beginning to acquire habits of thought conducive to the development of rich understandings and supportive of historical conversations, but it remains to be seen whether these will develop. Future research will explore these issues, and, in particular, the effects of specific classroom strategies for the use of the application on the development of historical thinking and classroom discourse. We are especially interested in the possible effects of task orientation and cooperative learning on students' development of historical thinking.

An unanticipated finding from the preliminary research also deserves further investigation. This is the suggestion that visual imagery might have had a mnemonic function in the students' mental representations of the material they covered. It seems possible that students used their memories of visual images to collect, organize, and cue their recollections of information about the people, places, events, and issues they explored. If this was in fact taking place, the phenomenon could, indeed should, be exploited in hypermedia design. The possibility clearly merits future investigation.

References

Booth, M. (1984) Skills, concepts, and attitudes: the development of adolescent children's historical thinking. *History and Theory, 22*, 101-117.

Comstock, G. (1978) *Television and Human Behavior*. New York: Columbia University Press.

Downey, M. T. (April, 1993) *Historical learning and the writing process: the Writing To Learn History project*. Paper presented at the Annual Meeting of the American Educational Research Association, Atlanta, GA.

Greenfield, P. M. (1984) *Mind and Media: The Effects of Television, Video Games, and Computers* Cambridge, MA: Harvard University Press.

Hammond, N. (1989) Hypermedia and learning: who guides whom? In H. Maurer (Ed.) *Computer Assisted Learning*. New York: Springer-Verlag, 167-181.

Heller, R. S. (1990) The role of hypermedia in education: a look at the research issues. *Journal of Research on Computing in Education, 22* (4), 431-441.

Howe, M. J. A., Ed (1983) *Learning From Television: Psychological and Educational Research*. Orlando, FL Academic Press.

Langer, J. A. (1992) *The Development of Critical Thinking in Academic Subjects*. Albany, NY: Center for the Learning and Teaching of Literature, University at Albany.

Nelson, W. A & Palumbo, D. B. (1992) Learning, instruction, and hypermedia. *Journal of Educational Multimedia and Hypermedia. 1*, 287-299.

Salomon, G. (1981) *Interaction of Media, Cognition, and Learning*. San Francisco: Jossey-Bass

Shapiro, A. M. & Spoehr, K. T. (April, 1992) *The Effects of Corpus Organization and Learning Goals on Learning from a Hypermedia System*. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Seixas, P. (1993) The community of inquiry as a basis for knowledge and learning: the case of history. *American Educational Research Journal, 30*, (2), 305-324.

Spiro, R. J. & Jehng, J-C. (1990) Cognitive flexibility and hypertext: theory and technology for the nonlinear and multidimensional traversal of complex subject matter. In D. Nix & R. Spiro (Eds.) *Cognition, Education, and Multimedia: Exploring Ideas in High Technology*. Hillsdale, NJ: Erlbaum, 163-205.

Spoehr, K. T. (April, 1992) *Using Hypermedia to Clarify Conceptual Structures: Illustrations from History and Literature*. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Swan, K. & Black, J. B. (1993) Knowledge-based instruction: teaching problem solving in a Logo learning environment. *Interactive Learning Environments, 3*, (1) 17-53.

The method to acquire the strategic knowledge on problem solving

Ryo TAKAOKA Toshio OKAMOTO

The Graduate School of Information Systems,
University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182, Japan
E-Mail: ryo@ai.is.uec.ac.jp

Abstract: The aim of this study is to construct an Intelligent CAI(ICAI) with the function of knowledge acquisition by the theory of Explanation-Based Learning(EBL). In this paper, we describe a learning system which acquires strategic knowledge by EBG algorithm. In particular, the mechanism of an inference engine, the reasoning process to generate an expert's explanation and the method to acquire strategic knowledge are discussed. Moreover, the method to apply the acquired strategic knowledge is proposed.

1 Introduction

In general, as a human goes on learning, his problem solving behavior is improved. One reason for this improvement is considered to be that as a human acquires the knowledge called "macro-rules" which make problem solving more efficient. It is not an easy task to formalize macro-rules as meta-knowledge in a domain knowledge base for Intelligent CAI(ICAI)[1,2,5]. Therefore, we have been developing the ICAI which automatically acquires the useful knowledge from the domain experts. As the experts give the learning system instructions of how to solve given problems, the system extracts the strategic knowledge of problem solving from the instructions by generalizing them. Then, the educational system teaches and coaches students, based on the acquired strategic knowledge. By replacing a student as an expert, that is, by making a student give instructions instead of an expert, the system would extract a student's strategic knowledge.

To realize these issues, the aim of this study is to design and construct an ICAI with the function of knowledge acquisition by the theory of Explanation-Based Learning (EBL)[3,4]. This paper describes the mechanism of machine learning system which can acquire strategic knowledge on problem solving from the instruction-sequence given by an expert.

Chapter 2 examines the asserted knowledge in the problem solving process. Chapter 3 discusses the method to acquire the strategic knowledge. First, the configuration of the learning system that automatically acquires the strategic knowledge is described. Secondly, the type of expert's instructions are analyzed. Then, we discuss how the system interprets the expert's instructions, and how the system extracts the strategic knowledge. In Chapter 4, we propose how to use the acquired knowledge.

2 The asserted knowledge in the problem solving process

The domain which is handled in this study is the tasks of physics (the question on the composition and decomposition of vectors) at a high school level. The problem solving process in this subject matter includes the following steps:

Step1: To understand the sentences of a given problem.

Step2: To add extensional semantics information related to the sentences of a given problem.

Step3: To make the appropriate mathematical expressions related to some parts in the structure of a given problem.

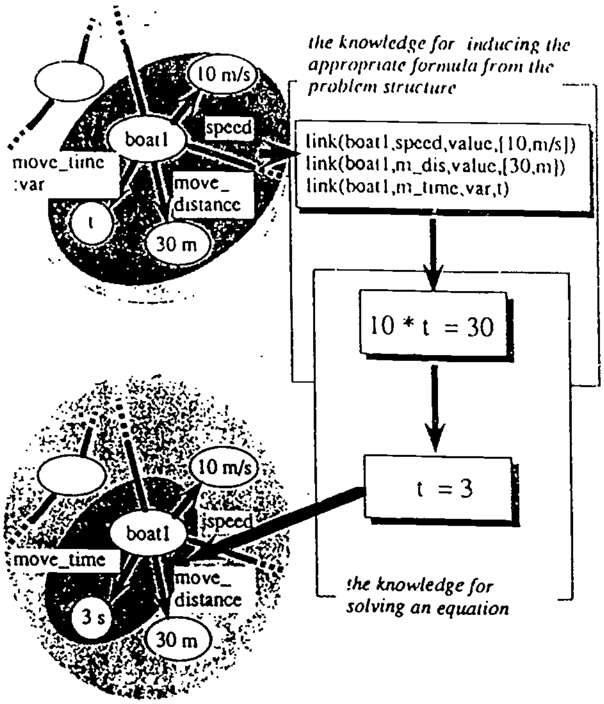


Figure 1: The relationship among the knowledge

```
basic_ope(70,
 [link(Obj1,S1,hensuu,[V,M/S]),
  link(Obj1,MT,value,[T,S]),
  link(Obj1,MD,value,[X,M])],
 [link(form5,relation,value,[S1,MT,MD])],
 [link(Obj1,S1,equ,[X=V*T,M/S])],
 []).

link(form5,relation,value,
   [sin_speed,sin_move_time,sin_move_dist]).
link(form5,relation,value,
   [sin_h_speed,sin_move_time,sin_move_h_dist]).
link(form5,relation,value,
   [sin_v_speed,sin_move_time,sin_move_v_dist]).
```

The knowledge for inducing the appropriate formula from the problem structure

```
basic_ope(203,
 [link(Obj1,Attri,equ,[X=V*T,U])],
 [],
 [link(Obj1,Attri,value,[V1,U])],
 [shiki(X=V*T,V1)]).
```

The knowledge for solving an equation

Figure 2: The internal expression for the domain knowledge

Step4: To solve the expressions, and calculate the values corresponding to the attributes of the objects.

We assume that the solution of a given problem is obtained by repeating the steps from Step2 to Step4. In order to construct the system which handles such a problem solving process, the following knowledge was prepared for the domain theory.

(1) The knowledge for understanding the structure of a problem.

(2) The knowledge for inducing the appropriate formula from the problem structure.

(3) The knowledge for solving an equation.

(4) The knowledge for solving plane trigonometry.

(5) The knowledge of a dictionary on the concepts/ technical terms.

Figure 1 shows the relations among (2),(3) and (4) in this system. The figure shows that the formulas related to three variables of speed, time, and distance can be applied to part of the problem structure, that is, "the speed of boat1 is 10 m/s" and "the distance to which boat1 moves is 50 m", and new information, that is, how long it takes for boat1 to move across the river is added to the structure of the problem. **Figure 2** shows the internal expression for the domain knowledge in this system.

3 The method to acquire the strategic knowledge on problem solving

3.1 The configuration of the learning system

When the leaning system is given a problem by an expert, the system learns how to solve it from problem solving statements specified by the expert. The system tries to find an appropriate equivalent operator-sequence by analyzing this process and generalizes them. Simply, the system extracts strategic

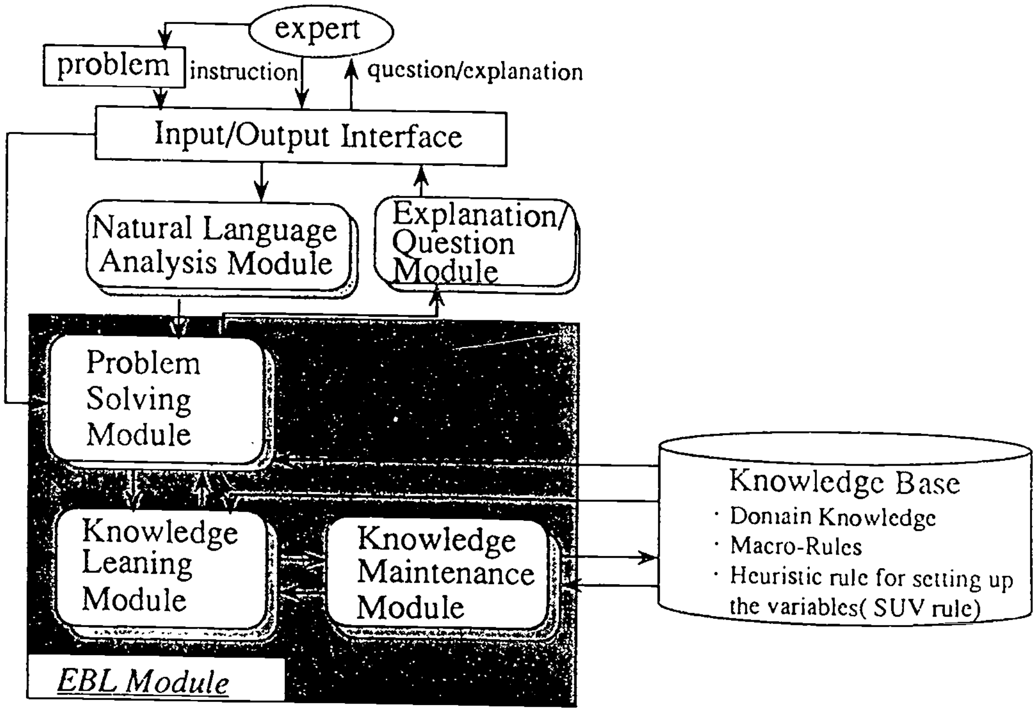


Figure 3: The configuration of the system to acquire the strategic knowledge

knowledge, which improves its problem solving performance, from expert's instructions. In this study, the strategic knowledge is the one that facilitates to solve problems efficiently with prediction, which is related to planning and intention. **Figure 3** shows the configuration of this learning system. The system consists of five modules and a knowledge base. "*Natural Language Analysis Module* " translates expert's instructions into the internal expressions which the system can interpret. "*Problem Solving Module* " builds up an explanation-tree. The instructions given by experts are classified into four different forms. Through the expert's instruction process, the system traces what kinds of knowledge is used by the expert. "*Explanation/Question Module* " has two functions. One is to explain what kinds of knowledge is used in expert's instructions. This function is displayed in a window on the screen. The other function is to confirm what attribute an expert asserts as a variable, to ask what its value means in the expression. This module is driven, whenever the system is given instructions for the former, and can't understand the expert's instructions for the latter. "*Knowledge Learning Module* " builds up a generalization-tree to acquire strategic knowledge. "*Knowledge Maintenance Module* " tunes up the acquired knowledge in the domain knowledge base.

This learning system acquires two kinds of knowledge. One is "**macro-rules**" acquired with EBG algorithm, and the other one is "**heuristic rules for setting up the variables**", which we call "**SUV-rules**". Heuristic rules for setting up the variables is the knowledge to set up attribute-value of an object, which the expert wants to know, as a variable in a particular situation.

3.2 The method to interpret Instructions

In this section, we describe how to construct an explanation-tree from a sequence of instructions given by experts. With a production system, the system tries to find the appropriate equivalent operator-sequence by analyzing an expert's instruction. The inference engine of the production system uses two kinds of working memories; the first one is to identify an instruction(ii_wm) and the second one is to generate an explanation-tree(ex_node). The ii_wms are used in a process to understand one instruction, when the system regards its instruction as the sub-goal. The ex_nodes represent a set of the knowledge, which have already known, in a state of the solving process. **Figure 4** shows an internal expression of ex_nodes in the system. The 1st argument of ex_nodes is a node number. The 2nd argument shows the

ex_node(1, link(boat1, move, value, river1), [],[],n,y).
ex_node(4, link(river1, sin_speed, value, [4, m/s],[],[],n,y).
ex_node(11, link(river1, give_an_effect,value,boat1),[1,4],[b,50],n,y).
ex_node(19, link(boat1, sin_speed,hensuu,[vh, m/s],[],[],hensuu,y).

ex_node(1st, 2nd, 3rd, 4th, 5th, 6th)

1st: a node number
2nd: knowledge
3rd: node numbers of precondition nodes in a list form
4th: an applied rule for the precondition nodes
5th: whether its knowledge is explained by experts or not
6th: the usablity in the next reasoning

Figure 4: The memory to generate an explanation-tree(ex_node)

knowledge about the attribute, its types of value, and the value of the object. The 3rd argument holds node numbers of precondition nodes in a list form. The 4th argument shows an applied rule for the precondition nodes. The 5th argument in ex_nodes shows whether its knowledge is explained by experts or inferred by the system itself. The 6th argument represents whether the Inference Engine (IE) can use this node or not from now on. For example, ex_node 11 in the figure 4 includes the following contents; the knowledge that "a river influences a boat" is resolved by using basic-rule 50 for 1 and 4 nodes, this node is inferred by the system in the process of identifying a given instruction, and IE can use this node in the next reasoning process. **Figure 5** shows the method to generate an explanation-tree. The system tries to convert the problem sentences into the appropriate internal expressions, which are translated by "Natural Language Analysis Module", as the initial state in ii_wm. At this time, the ex_nodes are a state of an empty list. When an expert gives instructions to the system, the system fills ii_wms with all information of the present ex_nodes and the expert's instruction. Then, the system regards the expert's instructions as the sub-goal, and understands its instructions by using a production system. The system continues to infer by means of the deep-first search algorithm. So, the system keeps a record of the intermediate hypothesis, until the system could find a given sub-goal. Moreover, to avoid a wasteful search, the system keeps the information of success or failure for each nodes. When the meaning of instruction is understood, the system adds the knowledge, which hasn't been ever included in ex_nodes, to ex_nodes in the forms of figure 4. The system repeats this cycles until experts inputs "finish the work",

We examined the types of instructions which experts asserted in the problem solving process on this subject matter. As a result, we found that the following four types of instructions are included in the explanation to solve problems.

(1) Instructions on attributes of objects, or its attributes and values.

(2) Instructions on defining attribute-value of an object as a variable.

(3) Instructions on a causal relationship among objects.

(4) Instructions on equations.

The system makes the knowledge processing corresponding to these four types of the instructions, and as a result of it, the system can identify the expert's problem solving process. The identification of the instruction types (1), (2), and (3) adopts the method mentioned above. The system has a function to transform the expert's equation into an equivalent equation when an expert gives the instruction type of (4) to the system. "Instructions on defining attribute-value of an object as a variable" is slightly different from the knowledge processing of (1), (3), and (4). Even though experts have the knowledge to set up attribute-values of an object as a variable in a particular situation, the system doesn't have it. So the system acquires the knowledge of SUV-rules from this instruction.

3.3 The method to acquire macro-rules

In this system, we explored the method of generalization with EBL algorithm. In general, a target concept to learn is given in the framework of EBL. When the system can solve a problem by expert's

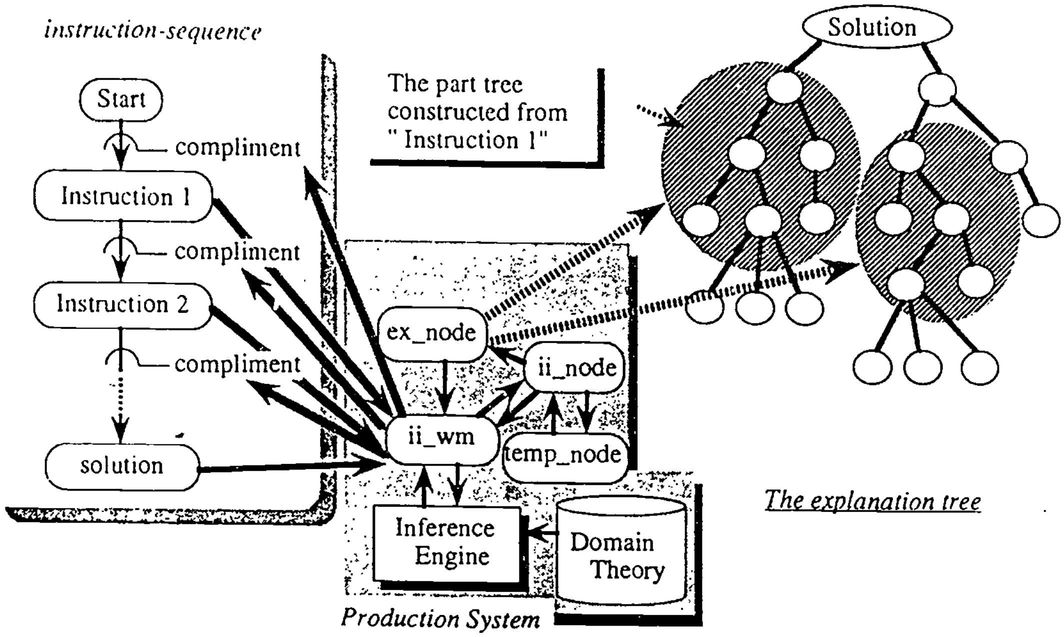


Figure 5: The method to construct the explanation-tree

instructions, the solution becomes the target concept. The given problem itself becomes a training example. The system regards one instruction given by the expert as a trigger to resolve sub-goals. Then, the instructions given by an expert work as bias, and the system constructs an explanation-tree. The system generalizes it within the domain theory. The operationality criterion in this system is the procedure which extracts the macro-rule from rule-sequences within the scope of causality. So the system can extract macro-rules from an example problem. **Figure 6** shows a problem and the example of instructions. **Figure 7** shows the example of the extracted macro-rule and SUV-rule.

4 Application of the acquired strategic knowledge

In this chapter, we discuss the meaning of acquiring strategic knowledge on problem solving with the EBL technique and the method of its application.

The learning system acquires the strategic knowledge from expert's instructions. This acquired knowledge is reflected as the template knowledge in a student model for an ICAI.

As mentioned in Chapter 1, by making a student give instructions instead of an expert, the educational system would extract a student's strategic knowledge. So, when the educational system diagnoses a state of a student model, and supports the student's learning activity, the strategic knowledge is reflected. By adding the acquired macro-rules to the system, the quality of explanation in the tutoring module is improved. If macro-rules are prepared for various states, then the system is able to give an explanation of solving for various similar problems. The system gives some rough explanations to the student who understands well how to solve problems, and details to the student who doesn't understand it well. Then, the system tries to select the more effective instructional strategy.

5 Conclusion

In this paper, we described the mechanism of a machine learning system which can acquire the strategic knowledge on problem solving from the instruction-sequence given by an expert. By using this system, the strategic knowledge of domain knowledge base for ICAI can be acquired as meta-knowledge, that is "macro-rules". Moreover, we referred to application of the knowledge acquired by EBL. Now, we have

PROBLEM

A boat with a speed of 4 m/s goes across the river with a speed of 3*root(3) m/s. It is 80 m in width. Now, the boat goes up the upper stream. Its angle is 30° for the flow direction of the river. Then, it arrives at the opposite side of the bank. Supposing that the position of the opposite bank locates at the position of b.

(1) In which direction does the position of b locate from the position of a and how far is it from the position of a to the position of b ?

ANSWER

Put the speed of vertical direction of boat as v_{1y}
$v_{1y} = 4 * \cos 30$
$v_{1y} = 2$

Put the speed of horizontal direction of boat as v_{1x}
$v_{1x} = 4 * \sin 30$
$v_{1x} = 2 * \text{root}(3)$

Put the time which boat moves as t
$2 * t = 80$
$t = 40$

Put the distance which boat goes forward horizon as x
$x = (3 * \text{root}(3) - 2 * \text{root}(3)) * 40$
$x = 40 * \text{root}(3)$

The boat arrives at the point of downstream with 40*root(3) m far from the position of a

Figure 6: The example problem and the example of instructions for its problem

```
macro_ope(2,
 [link(Obj1,sin_dir,value,_802),
  link(Obj2,give_an_effect,value,Obj1),
  link(Obj1,com_move_time,value,[_834,S]),
  link(Obj1,sin_h_speed,value,[_866,M/S]),
  link(Obj1,com_move_h_dist,hensuu,[_890,M]),
  link(Obj2,sin_h_speed,value,[_922,M/S]),
  link(Obj1,com_h_speed,hensuu,[_954,M/S]),
  link(Obj2,_974,value,+)],
 [link(dir_h,relation,value,[sin_dir,_974,_802,-]),
  link(com_vh_veloc,relation,value,
     [_974,sin_h_speed,_1076,com_h_speed]),
  compare(_866,_922),
  shiki(_954=_922-_866,_1126),
  link(form5,relation,value,
     [com_h_speed,com_move_time,com_move_h_dist]),
  shiki(_890=_1126*_834,_1178)],
 [link(Obj1,com_move_h_dist,value,[_1178,M])],
 [[[b,160],link(Obj1,_974,value,-)],
  [[b,100],
   link(Obj1,com_h_speed,equ,[_954=_922-_866,M/S])],
  [[b,205],
   link(Obj1,com_h_speed,value,[_1126,M/S])],
  [[b,72],
   link(Obj1,com_move_h_dist,equ,[_890=_1126*_834,M])],
  [[b,203],
   link(Obj1,com_move_h_dist,value,[_1178,M])]]).
```

(a) Macro-rule

```
suv_ope(1,
  [link(Obj1,sin_speed,value,[_762,M/S]),
   link(Obj1,sin_angle,value,[_786,D])],
  [make_vaiable([Obj1,sin_v_speed],_892)],
  [link(Obj1,sin_v_speed,hensuu,[_892,M/S])],
  []).
```

(b) SUV-rule

Figure 7: Macro-rule and SUV-rule

been constructing an ICAI based on the theory of EBL[6]. As the next research step, the examination of an operationality criterion to select the acquired macro-rules, and the development of three-agents interaction system for ICAI by using the acquired knowledge must be explored.

References

[1] Claude.F., and Gilles.G. INTELLIGENT TUTORING SYSTEM: At the Crossroad of Artificial Intelligence and education. Ablex Publishing Corporation Norwood, New Jersey, 1990.

[2] E.Wenger. Artificial Intelligence and Tutoring Systems. Morgan Kaufmann, Los Altos, CA, 1987.

[3] Dejong.G., and Mooney.R. Explanation-based Learning: An alternative view. Machine Learning, 1(2), 145-176, 1986.

[4] Mitchell.T., Keller.R., and Kedar-Cabelli.S. Explanation-based generalization: A unifying view, Machine Learning, 1(1), 47-80, 1986.

[5] Yamada.S., and Tsuji.S. Generation of Instructional Knowledge with Learned Strategy Knowledge. JSAI, 5(6), 787-795, 1990.

[6] Takaoka.R., and Okamoto.T. Muti-Interactive Learning Companion System with EBL. The 9th Conference On JET, 1993.

On the design and development of Pedagogy-First CAI tools for CS Education

K. Vadaparty H. Salem W. Thompson S. Mathew
Department of Computer Engineering and Science,
Case Western Reserve University, Cleveland, OH 44106

R. Molmen
Baldwin-Wallace College
Berea, OH 44017

Abstract: In this paper we present the results of an ongoing project on the *design and development* of multimedia instructional material for teaching and learning a number of computer science topics at both graduate and undergraduate levels. We emphasize on pedagogical requirements such as *dynamic* animation of the changes in the data structures induced by the user-input, *dynamic* automated examinations that are *not* multiple-choice types, incorporation of meta-algorithmic information, the ability to step through algorithms both forward and backward, etc. We divide the subject-matter into *themes* and develop the software for each of these themes. The current theme we considered is tree data structures such as binary trees, red-black trees, 2-3 trees, etc. We believe that our tools relieve the students of the burden of visualizing the data structures and also enable the teachers to concentrate more on the principles behind the data structures than on the mechanical operations. We believe that our project is one of the first to exploit the sophisticated PC tools for CAI.

1 Overview

In this paper we present the results of an ongoing project on the *design and development* of multimedia instructional material for teaching and learning a number of computer science topics at both graduate and undergraduate levels. One of our *fundamental goals* in this project is in developing computer aided instruction software *that meets a number of important pedagogical* requirements. We believe that the progress of the project so far, as reported in this paper, *makes a significant first step in achieving this goal.* There are a few questions one needs to ask regarding CAI tools. First, is there a need for more CAI tools. Our answer is a definite "yes". In brief, our position on these systems is the same as expressed in [NVH86]:

"During the 1970's a second wave of CAI systems [started]. Despite significant government and commercial commitments to large-scale CAI projects, the record of actual use has been spotty ... recent proliferation of "smart" machines capable of instructional ... [with graphics capabilities] ... and above all, affordable prices... has opened new areas in which ... [it is possible to operate with] "learning-by-doing" paradigm... These ad hoc, "do-it-yourself" uses of CAI techniques will ultimately succeed where frontal attacks have failed ..."

Powerful hardware and software tools in the PC world have never been more affordable than now. We believe that our project will start new developments that hopefully exploit the power of modern PCs in computer science education.

The second question is who should develop CAI tools: since these tools should "distill" the experience of the teachers (and the difficulties of students), we strongly believe that CAI tools (prototypes at any rate) should be developed by teachers and students, although they may be marketed by publishers or the software industry (perhaps after adding more features). Finally, the third question is what should the level of genericity be. We believe that computer aided instruction ought to divide the subject matter into *themes*, just as we do in classroom instruction. A theme is a family of lessons that teach a particular algorithmic or structural paradigm. Examples of themes abound: *tree data structures, graph search algorithms, sorting, sweepline paradigm, divide and conquer paradigm*, etc.

The main contribution of our work is in *designing and developing* computer aided instruction software for the tree data structures *theme*: it satisfies a number of important pedagogical requirements as described in Section 2. Furthermore, it is *readily usable* by teachers (students) for teaching (learning) tree data structures. A student or teacher can run (client configuration) our system on MS-Windows at almost minimal configurations.

The paper is organized as follows. Section 2 describes the desired pedagogical requirements that any CAI system should satisfy; most of these requirements are satisfied by our system. Section 3 briefly surveys the related work and provides a comparison with some of the major CAI efforts. The paper concludes with screen-captures of actual running examples.

Implementation: The project, implemented in Visual Basic 3.0, has about 32,000 lines of code and took about 3.5 person-years.

2 Pedagogical requirements that our system satisfies

In this section we describe some of the important pedagogical requirements that any CAI software for computer science lessons should satisfy. Note that our software satisfies most of these as shown in the illustrations later. Some of the issues are specific to the *theme* of tree data structures while some may be useful for other themes as well. To the best of our knowledge, many of these requirements are not met by any of the CAI software currently available; textbooks certainly do not come even close to meeting these. Most of these requirements are currently met by the efforts of enthusiastic instructors, teaching assistants, and sometimes students themselves. The following discussion refers to a number of tree data structures which can be found in classical references such as [CLR90, Tar83]. Our listing here is not intended to be exhaustive and exclusive; it is the best we could get from our own experience and the experience of others [Nie90, NVH86, EH92, Fla87, Sto89].

1. *Interactive Animation*: When using tree data structures, one of the most important issues is the change in the topology of the tree when a new value is inserted or deleted. This may involve one or more rotations (AVL trees, red-black trees, splay trees), splitting (2-3 trees, B-trees, etc), change of attribute-values of the tree-nodes (change of color in red-black trees, values in order-statistics trees), and so on Any software purported to aid the instruction about these tree data structures should animate the changes of the tree topologies. Furthermore, such an animation should be *user directed*.

2. *Stepping Through*: It is of great help to the student see how the updates algorithms (delete/insert) modify the tree data structures step by step. This involves coordinating the text of the algorithm, the explanation of what is happening at the current step and the interactive animation of the tree while the current step is being executed

3. *Superimposition of Meta-algorithmic Information*: When a tree topology changes due to updates, it follows certain rules. These rules are specified through ***canonical cases***. For instance, in the case of red-black trees, [CLR90] describes three canonical cases for insertion into red-black trees. A user-defined example must show which canonical case that it is using to perform the current rotation, etc. Note that, strictly speaking, this superimposition is not a part of the algorithm; it is "meta-algorithmic" information.

4. *Pathological Examples* that are **dynamic**: We observed in our teaching and learning that it is always good to present "killer examples", i.e., examples that are absolutely bad for the particular algorithm or data structure. It is, for example, highly instructive to show an order of insertions which causes a skewed binary search tree and then show how this very order of insertions produces a balanced red-black tree. Furthermore, these examples should be dynamic, that is, different at different runs of the software.

5. *Dynamic Quizzes*: An important feature of learning process is also "doing". We believe that one way to help the students participate in the understanding process is to make them answer questions. The system should be able to "create" dynamic quizzes. Note that these should not be just multiple-choice questions, but should require user participation that is more involving: say, (i) insert/delete (ii) perform the consequent fixup steps (rotations) in a red-black tree, etc. This type of user participation increases understanding of the subject matter better [Sto89, Fla87].

6. *Augmenting Data Structures* with procedures: One should have procedures added to the nodes of a data structure to evaluate certain properties. For instance, each node of a tree can be augmented with a procedure that computes the *predecessor* (or *successor*) of the node.

3 Related Work and a Comparison

Although there is an abundance of software for teaching concepts at the K-12 level of education, there are very few popular software packages for teaching Computer Science lessons at the university level. An excellent description of ups and downs of CAI from sixties through eighties is presented in [NVH86]. Some of the examples of CAI tools are PLATO whose derivative was AUTOTOOL[Sto89], and BALSA and ZEUS [BS84. Bro92]. In principle, we could have used Zeus to develop lessons on various themes (as is done for red-black trees [Hey93]). However, we opted not to use Zeus for a number of reasons. We observed that considerable amount of work needs to be done to develop teaching tools that can be readily used. For example, labeling nodes, indicating canonical cases, etc., requires special view development and coding into the annotations. We believe that using sophisticated software packages and inexpensive graphics hardware available for PC's one can develop the desired instructional software from scratch with the same or less effort. This belief is bolstered by our development of a number of data structures. Note that Zeus is a general purpose algorithm animating environment, and was not developed only for educational tools; unlike Zeus, our software was developed for educational purposes only, and can not be used to animate arbitrary algorithms.

Acknowledgements Kumar Vadaparty would like to acknowledge the partial support of the Lilly Foundation for the Fellowship award which included a semester's release time. He also would like to thank Dr. Ray Neff, Dr. Sandra Russ, Dr. Tom Kicher and Dr. Lee White. As a team, we would also like to thank Mr. Jamie Carl, for the initial work of developing binary search tree animation using Turbo Pascal. Thanks go to Ms. Dawn Cohen for pointing out the idea of dynamic quizzes.

References

[Bro92] Brown, Marc H. Zeus: A system for algorithm animation and multi-view editing. SRC Research Report 75, DEC, February 1992.

[BS84] Brown, Marc and Segewick, Robert. A system for algorithm animation. *Computer Graphics*, 18(3):177–186, 1984.

[CLR90] Cormen, T., Leiserson, C., and Rivest, R. *Introduction to Algorithms*, volume 1 of *Computer Science*. McGraw Hill, 1990.

[EH92] Edwards, D.N.A and Holland, S. *Multimedia Interface Design in Education*, volume 76 of *Computer and System Sciences*. Springer-Verlag, 1992.

[Fla87] Flanders, H. Mathematical Calc. Education. In *AMS (summer) Minicourse, Salt Lake City, Utah*, 1987.

[Hey93] Heydon, Allan. Search tree animation. Technical report, DEC, 1993.

[Nie90] Nievergelt, J. Computer Science for Teachers: A Quest for Classics and How to Present Them. In *Proceedings of the second international conference on computer assisted learning*, 1990.

[NVH86] Nievergelt, J, Ventura, A., and Hinterberger, H. *Interactive Computer Programs for Education-Philosophy, Techniques, and Examples*. Addison-Wesley, 1986.

[Sto89] Stone. M.G. Interactive COSTOC Tutorials. In *Proceedings of the second international conference on computer assisted learning*, 1989.

[Tar83] Tarjan, R.E. *Data Structures and Network Algorithms*, volume 1 of *Applied Mathematics*. CBMS-NSF, 1983.

Examples

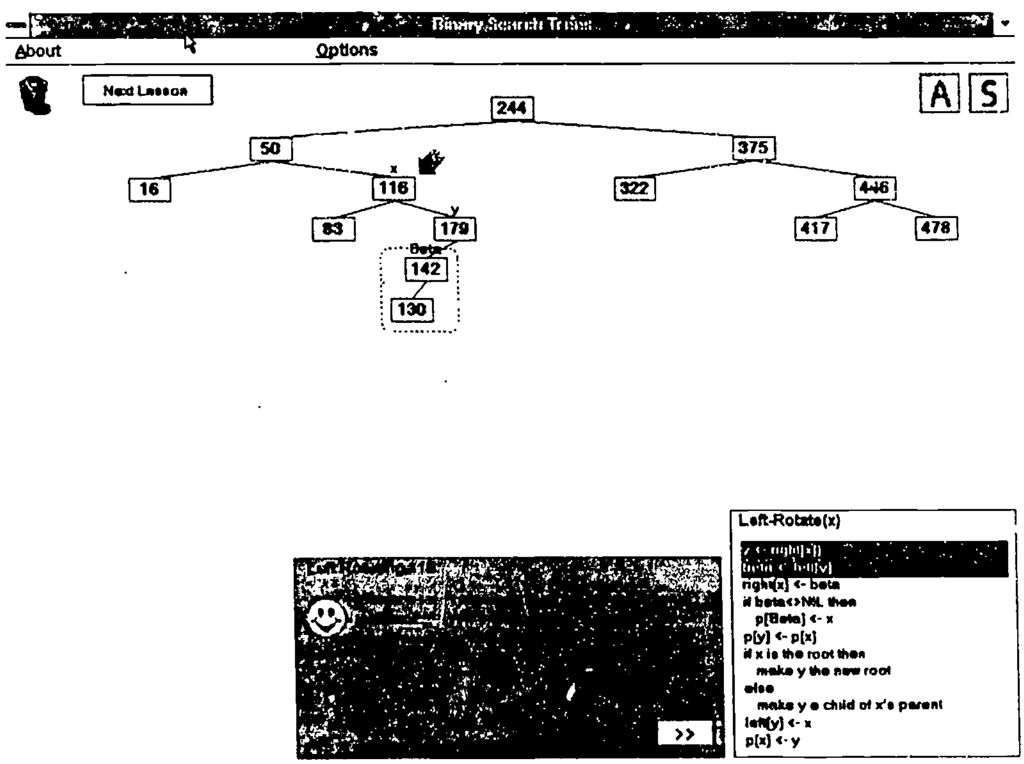


Figure 1. An example of a Binary Tree

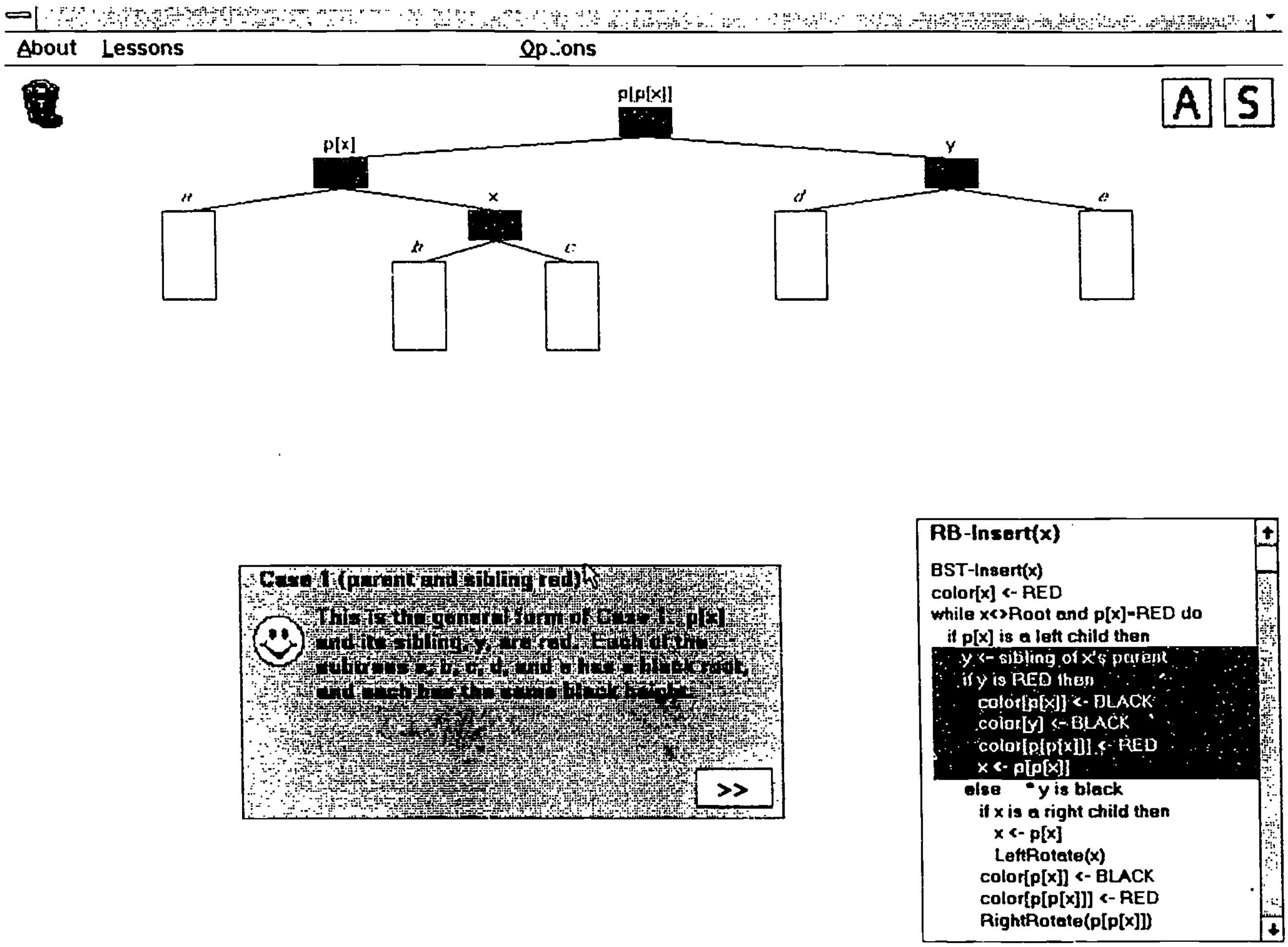


Figure 2. Canonical Case of Red-Black Trees.

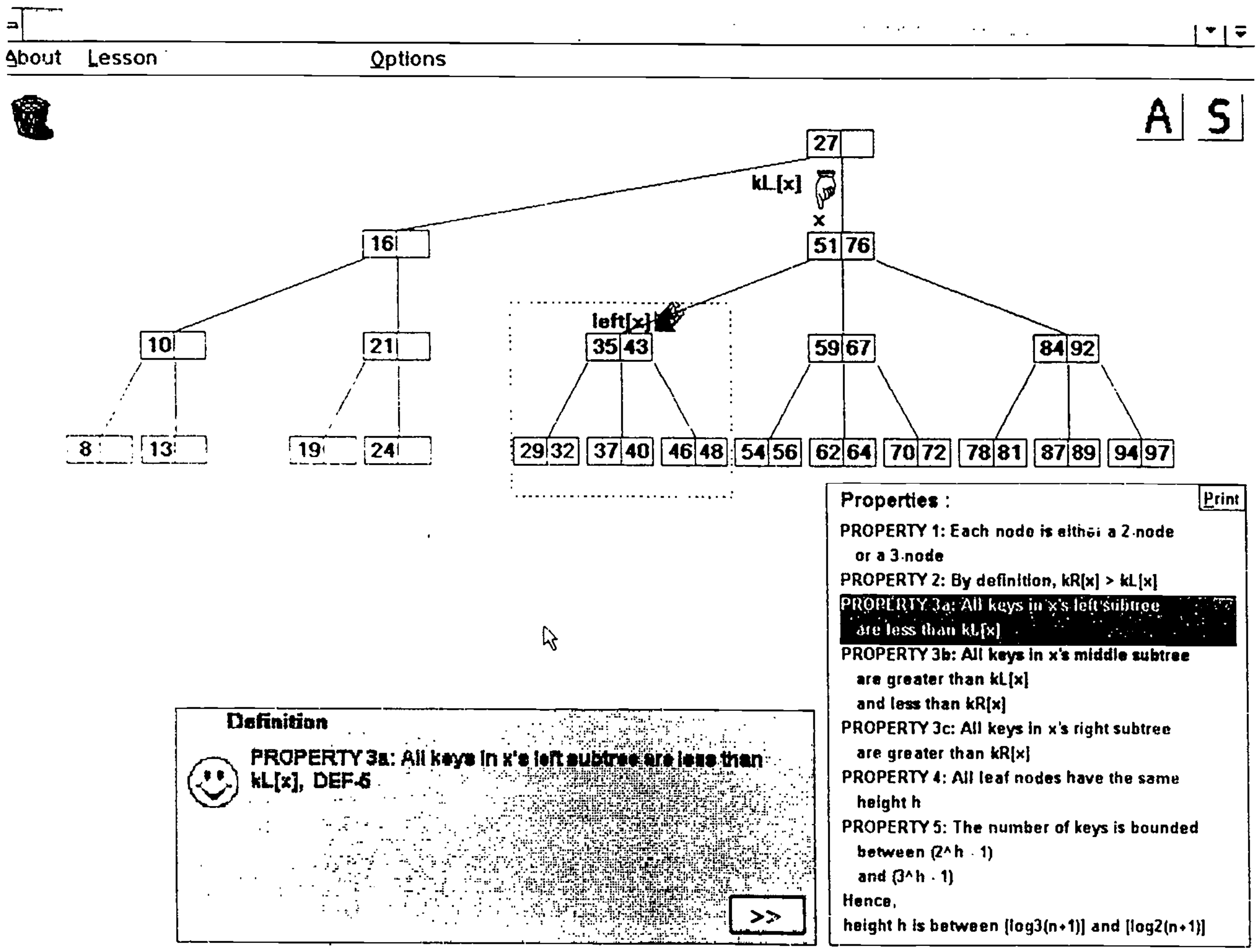


Figure 3. Example of a 2-3 Tree.

The Princess Bride™ : Letting the Resources Drive the Instruction

JOANNE WALKER
Department of The Language Centre
University of Victoria
Victoria, B.C., Canada V8W 3P4
E-mail: LANGCEN@UVVM.UVIC.CA

Abstract: This paper presents an example of the process used by our research and development team of evaluating technology and resources for use in Computer Aided Language Learning (CALL). One evaluative process is to create a real program then submit it to potential users – students and instructors – for assessment and criticism. In this case the team creates an interactive multimedia program for ESL using a Hollywood movie as a resource and, at the same time, tests an authoring package and experiments with a new development platform.

Introduction

The pitfall of letting the latest in glitzy technology become the driving force in Computer Assisted Instruction or Learning (CAI/CAL) is clearly articulated in the advice given by leaders in the field. Schwier and Misanchuk caution, "...as every instructional designer knows, the power of an instructional approach lies in the science and art of instruction – not in the technological components which comprise a hardware system" (Schwier and Misanchuk, 1993). However, there are situations in which a new technology permits access to materials for instruction that were heretofore inaccessible or difficult to incorporate into the teaching process. In these situations technology can and should inspire users (Ervin, 1993). In research and development it is not at all unusual to find the resources driving the instruction, a sort of we-have-a-new-CD-ROM-drive, let's-use-it-to-teach-French-verbs approach. This approach can be modified to an acceptable one if the new technology is applied in such a way as to incorporate sound instructional methods.

Research and Development personnel of the University of Victoria Language Centre often initiate software development in the form of prototype or sample programs in order to test new hardware and software or to demonstrate what is possible with the newest computing technology. In such projects, we choose the computing language, natural language, subject and platform. Occasionally the programs we have developed are immediately usable by instructors and students, but most often they provide template resources for other programs which are produced at the initiation of and in collaboration with foreign language instructors.

The Princess Bride™ : and adventure in ESL is such a project. It was motivated by the availability of the movie on videodisc; our need to test the authoring program ToolBook™ and to experiment with multimedia on the DOS/Windows™ platform; the continuing search for new resources for language teaching; and the desire to interest ESL instructors in Computer Aided Language Learning (CALL).

Video is recognised as an excellent resource for language instruction (Terrell, 1993). Incorporating video into CALL means that the computer must be able to control the video device. While it is possible for the computer to control a VCR, it is unsatisfactory because of the inherent shortcomings of tape as a medium. VCR tapes have slow access time and sequential frame format. When tapes stretch, the timing codes must be updated. Videodiscs, on the other hand, are fast, almost indestructible, and each frame is immediately and directly accessible by the computer. At the present stage of development at the Language Centre, purchasing someone else's videodisc is preferable to producing our own. With increased consumer popularity of videodiscs, Hollywood movies are now often available in videodisc format. Therefore, we decided to investigate the possibility of using Hollywood movies as video resources for multimedia ESL software. "Repurposing" a videodisc in this way is not without peril, as Schwier and Misanchuk warn us in their excellent text on producing interactive multimedia: "This [repurposing] is an imposing task, and the developer must be particularly sensitive to constructing interaction, remedial segments, and effective feedback into the CAI, because the video will likely not be constructed interactively. If this caution is ignored, you run the risk of developing segmented, linear video attached to a page-turning device" (Schwier and Misanchuk, 1993).

The Process

The remainder of this paper examines the phases of producing an interactive multimedia program using *The Princess Bride* ™ videodisc.

Phase 1

The first step in the project development was to transcribe the script and to divide the movie into workable sections. Since *The Princess Bride*™ encompasses four sides of two CAV videodiscs each side conveniently became a chapter in the program. Each chapter was divided into the recommended three-to-four minute segments (Stanley and Griffin-Castro, 1992) called "episodes". Some episodes are as long as five minutes because they contain action scenes with little dialogue. Incidentally, it is important to note that the videodisc must be in the CAV (Constant Angular Velocity) format in order to use still-frame clips.

Since our potential customers, ESL instructors and students, have had little or no experience with, or interest in CAI, the initial programming was done without consultation with the end-users. I believed that it would be easier to get students and instructors involved in the project if we had something already prepared to show. Such production without consultation is possible in the Language Centre only because my background includes teaching and instructional design.

The first six episodes use one side of the videodisc. After programming these into various viewing, comprehension and grammatical exercises, I showed other Language Centre personnel, ESL instructors and ESL students the program and asked for their comments and suggestions on the work so far. At first they were impressed with the technology and the program, but soon they began pointing out the shortcomings. The program content was too complicated for Beginning and too simple for Intermediate ESL students. The students did not have enough control of the program. There was not enough use of the spoken language portions of the movie. I had not taken into consideration that most ESL students cannot type and they certainly are not good English spellers. By not programming for the two-button mouse, I had even ignored the possibility that one of them might be left-handed.

Once they understood that they would not have to become computer programmers, two ESL instructors agreed to help with the content and methodology components of the program, as time permitted. Language Centre staff vetted the program for bugs, inaccuracies and user-identified shortcomings. Another Language Centre software developer helped me with the content and structure and a programmer friend wrote some of the more difficult computer code.

Phase 2

The ESL instructors made many useful suggestions. For example, they suggested that I refer to the study guides of a local college (Camosun, 1988) to establish that the language and complexity of the movie was appropriate for upper Intermediate level students.

For the video presentation, we attempted to follow Tomalin's "The three stages of a lesson using video" (Tomalin, 1986), in which each episode was studied in three sessions. First the student would view the episode straight through and do a comprehension exercise. Second, he or she would view the episode again with access to pause and replay options plus a copy of the episode script with "hot key" vocabulary words. The student clicks the mouse on a word to see its definition and an example sentence. For Tomalin's third stage, "extension and transfer" we used a series of exercises.

The content of these exercises depended on what happened in the movie and included listening comprehension, order of events, word order and vocabulary building. Grammar exercises concentrated on such topics as English idioms and expressions, phrasal verbs, relative clauses, positions of modifiers and verb tenses. Exercises were in the form of multiple choice, fill in the blank and manipulating objects. Keyboard entry was kept to a minimum.

Instead of navigating the program linearly with previous and next buttons, we used a main menu with a list of episodes that gave the users access to the episode menus. The user moved around in an episode from a pull-down menu in the menu bar. While free access to all parts of the program was not prohibited, it was was not encouraged. We wanted the user to view a segment, perform a comprehension task, view the segment again then do the exercises. At the request of one ESL instructor, we built in the ability to keep track of the movements of the user, the time spent on any one portion of the program, and how many questions were attempted in each exercise. We thought that his would also show us how easy, difficult, or popular any one part of the program was.

After the incorporation of initial suggestions, we demonstrated the program, still covering only the first side of the videodisc, to two intermediate ESL classes. We put the program into our drop-in multimedia lab so students could use it when they wished. Initial interest was high but dropped off quickly. Most students watched the six episodes of the first side of the videodisc, poked around in the exercises a little, then watched the rest of the movie using a remote control. Obviously we had ignored the advice of Schwier and Misanchuk – we had developed a segmented, linear video attached to a slow page-turning device.

We are not disheartened by these results. Our own observations had shown that the program was difficult to navigate and depended too much on computer instruction in a language the user was trying to learn. It was boring, there was not enough feedback or interaction and worst of all, it was not an appropriate use of the resource – the movie. Since understanding a foreign language when it is spoken in normal conversation by native speakers is a goal of foreign language instruction (Terrell, 1993), the goal of the program should be listening comprehension and vocabulary building. It should give the student the opportunity to hear and learn English in context as spoken by natives.

We did realize that until an ESL instructor became a full collaborator in the program and was willing to make it a part of his or her curriculum, the pedagogical content of the program might not complement a particular course curriculum.

Phase 3

After further research into how to teach using video and how to use video in interactive multimedia, the third rewrite of the program is underway. We have resolved to identify and use non-verbal signals in the movie that will aid the student in understanding the language (Bini, 1993) and to make the program more interactive and to make the interaction emulate interpersonal communication (Schwier and Misanchuk, 1993). Since we have just purchased a top-of-the-line graphics package (once again letting the resources drive the instruction), we plan to use graphics and animation to replace much of the on-screen instructional text. We want to design a more user-friendly navigation system and create an iconography. Iconography in a computing context means the consistent use of icons to provide instant visual clues to what the program requires from the user, how the user navigates the program, what the purpose is of each screenful of information and what user interaction is available at that stage of the program.

Conclusion

The University of Victoria English Language Program teaches between 800 and a 1000 students each year. A majority of those students are modern young people from highly technical cultures who are not only undaunted by the technology, but have come to expect it – a phenomenon that is present in a growing number of foreign language classes across North America. Movies are a part of that technical culture. They contain entertainment, instruction and example, and are a comfortable subject for social conversation. As a resource for teaching English, movies provide examples of cultural and verbal exchanges that do not appear in classrooms. By incorporating movies into our instructional technology, we can present the user with a learning package that is familiar and comfortable. As a resource it is reusable, non-judgmental, infinitely patient, academically sound and entertaining.

References

Camosun College. *English As a Second Language Study Guide*, Upper Intermediate. Camosun College, Victoria, B.C. 1988

Bini, Milena. Films as a Pedagogical Aid in the Teaching of a Second Language. L2 and Beyond. *Teaching and Learning Modern Languages*: 1993: 270 - 271.

Ervin, Gerard L. Can Technology Fulfill Its Promise. *The IALL Journal of Language Learning Technologies*, Vol. 26 No. 2 Spring 1993: 13 - 14.

Schwier, Richard A. and Misanchuk, Earl R. *Interactive Multimedia Instruction*. Educational Technology Publications. Englewood Cliffs, NJ, 1993: 3, 9.

Stanley, M. and Griffin-Castro, P. Video at All Levels. *The Proceedings of The Second International Conference on Foreign Language Education and Technology*: 1992: 191 - 198.

Terrell, Tracy David. Comprehensible Input for Intermediate Foreign Language Students Via Video. *The IALL Journal of Language Learning Technologies*: Vol. 26 No. 2 Spring 1993: 17 - 18.

Tomalin, Barry. *Video. TV and Radio in the English Class: An introductory guide*. London Macmillan Pub. Ltd. 1986: 21 - 22.

ELM-PE
A Knowledge-based Programming Environment for Learning LISP

GERHARD WEBER AND ANTJE MÖLLENBERG
Department of Psychology
University of Trier, D-54286 Trier, Germany
E-Mail: weber@cogpsy.uni-trier.de

Abstract: The knowledge-based programming environment ELM-PE is designed to support novices learning the programming language LISP. It has several features that are especially useful in learning to solve problems in a new, complex domain. ELM-PE is an open, complete programming environment. It offers several facilities, e.g. visualization of executing programs, example-based programming, and cognitive diagnosis of program code, that go far beyond traditional ITSs. In an evaluation study, ELM-PE proved to facilitate learning. Students performing well in a final exam used the helping facilities more often during the first lessons of learning programming.

Novices in programming face many problems affecting their learning process and programming success. Learning to program includes acquiring knowledge and skills in various domains: Using the programming environment, learning a programming language's syntax and semantics, understanding a problem and translating it into an executable plan, developing algorithms and programs, and testing and debugging programs. In order to aid novices develop such abilities, various systems have been devised. These comprise Intelligent Tutoring Systems (ITS) (e.g., the CMU-LISP-Tutor (Anderson, Conrad, & Corbett, 1989), or BRIDGE (Bonar & Cunningham, 1988)) as well as knowledge-based programming environments and help systems (e.g., PROUST (Johnson, 1986), SCENT (McCalla & Greer, 1993), or ABSYNT (Möbus & Schröder, 1993)).

The knowledge-based programming environment ELM-PE is designed to support novices learning the programming language LISP. It has several features that are especially useful in learning to solve problems in a new, complex domain. These features are designed to avoid unnecessary mistakes, to give immediate feedback, to reduce working memory load, to support learner activity, and to support example-based learning. In contrast to an ITS like the CMU-LISP-Tutor, ELM-PE is an open, complete programming environment. Students are allowed to work on exercises with and without help from the system, to choose exercises on their own, to plan, to program, and to debug function definitions, and to ask the system for help if they need it. Students are allowed to produce programming errors, because students learn from debugging and from reflecting on their bugs and misconceptions. Therefore, debugging and helping facilities specifically designed for beginners are integrated into the programming environment. The system remains passive as long as possible and offers advanced help only on demand. Immediate feedback is given only for handling the graphical user interface, for syntactical errors, and if the student wants to work on a further exercise while the solution to the previous task is still incorrect.

The ELM-Programming Environment

ELM-PE can be run in three modes (LISTENER, EDITOR, and EXERCISE) among which the user can switch at any time. Modes differ according to the amount of support provided, so user control over the learning process is increased. For all modes passive as well as active help tools are implemented, using static as well as dynamic knowledge. Passive tools ensure the software-ergonomic principles of user definability and flexibility by leaving control to the student. Active tools interrupt students, but only when unnecessary errors (generally an attempt to use syntactically wrong input) or really serious errors (like switching to the next exercise when the actual solution is still incorrect) are detected that need to be dealt with immediately. This concerns the design rules of tolerance and transparency. Basic to the system is the Listener mode. Each following mode inherits all features introduced in the previous mode. Users are thus able to choose the level of specific support they wish. The elaborated help system and the system's flexibility make us believe that ELM-PE exceeds most traditional programming environments in assisting beginners.

To demonstrate the scope of the system some of the facilities of ELM-PE supporting beginners learning a new programming language will be sketched briefly here:

Listener Mode

The system always starts in Listener mode which is the least supportive way of interacting with ELM-PE. In listener mode, syntax-structuring facilities are not activated. Transparency is enhanced by dimming all menus which are not applicable in the current status of the system and by system messages in the status line at the bottom of the screen. Messages comprise hints about what the user is expected to do next, short feedback if an action was rejected or information about the time the system will take to perform some task.

All messages from the system displayed in the status line and all error messages displayed in the LISP listener can be explained. This is very helpful for novices because in the beginning most error messages may be more confusing than useful. For common and very often occurring errors a short description is given why such errors can happen. When running in Editor mode, that part of the code where the run-time error occurred is also highlighted in the program window. For every object visible on the screen, further information may be retrieved via special mouse clicks, so passive on-line help is available during all stages of working with ELM-PE.

Editor Mode

In Editor mode, most supporting features of the system are available, e.g. programming in a syntax-driven structure editor, using examples, visualizing the evaluation of LISP-expressions, and displaying that part of code of a function definition where an error was detected.

Structure editor. Coding function definitions is supported by a syntax-driven LISP-editor (Köhne & Weber, 1987) to reduce syntax errors. In the program window of the structure editor, LISP-code can be produced by filling in slots for LISP-expressions with syntactically correct LISP-code, or with schemata for function calls containing slots for their arguments. These schemata can be accessed from a menu of LISP-functions, or from typing in the first part of a function call beginning with the function name. Additionally, code can be inserted from a buffer showing the most recently typed in or deleted expressions or by directly typing in complete LISP-expressions in the input window. The structure editor accepts as input only code that would result in syntactically correct LISP-expressions. Otherwise the input is rejected and an explanation of why the input would result in an incorrect expression is given.

Example-based programming. Examples of LISP-code can be displayed in a separate example window to support example-based learning (Neal, 1989). Examples can be selected from examples discussed in the learning materials of the LISP-course as well as from the set of function definitions the student has coded on its own. The example window has the same functionality as the program window of the structure editor, so LISP-expressions can be altered and parts of the code can be copied into the program window.

Visualization. One central principle in learning programming is to visualize the flow of data during evaluation of programs. This is supported by a stepper showing all evaluation steps. Corresponding expressions from self-defined function definitions – including definitions of sub-functions – are indicated in the program window. The stepper is supposed to help students envision dynamic program properties and to encourage active and self-directed learning. Additionally, those parts of the program code that are responsible for errors or are explained by the knowledge-based diagnosis can be highlighted in the program window.

Exercise Mode

The highest level of support is given in Exercise mode. When working in this mode, students select a programming problem to be solved from a menu of exercises for the lesson he or she is working on. The description of the task and some input-output-examples are displayed in a separate text window. In addition to all other facilities, special active as well as passive help can be given in this mode.

Dynamic analysis. If work at a task is aborted and the student has completed the code, i.e., no slots remain, the code is analyzed dynamically. A list of input-output descriptions is used to determine whether the student's code returns the expected result. In case this is not true, the student is informed in a dialogue box and asked whether he or she still wants to terminate work. This is one of the few situations where the system is active by its own and will interrupt the student. Based on the input-output pairs the system used to test the function definition, the system is able to offer a function call with arguments leading to an incorrect result. Thus, students are encouraged to try to debug programs on their own. This is important because frequently students omit a thorough testing phase. In previous programming cou ses we observed that functions often were tested only once. If the arguments used do not reveal the error, students mistakenly think their solution was correct. Therefore, for this analysis to work the list of input-output descriptions must be complete so every error can be detected. The goal of this tool is to point out to students that extensive testing is an important skill as it can yield information about the type and location of errors.

Automatic cognitive diagnosis. Students can ask the system for help if they do not know how to complete coding the function definition, if they are not able to find an error in the code, or if they simply want to know whether their solution for the programming task will be correct. To offer students a complete explanation of errors and suboptimal solutions and to explain which plans have to be followed to solve the task correctly, a "cognitive diagnosis" was developed as a central "intelligent" component in the ELM programming environment. This diagnostic tool is based on the episodic learner model ELM (Weber, Bögelsack, & Wender 1993). The diagnosis results in an explanation how the program code submitted to the cognitive diagnosis could have been produced by the learner. This explanation is the basis of the system's hints to the learner what part of the code is incorrect, which plans must be followed to solve subgoals during problem solving and possibly partial solutions in the context of the learner's solution. Explanations are stored in the individual, episodic learner model. They can be used as the basis for retrieving structural and semantic analogies (Weber, 1991). As this component is central to the knowledge-based programming environment, it will be described in some more detail in the next section.

ELM: A Case-based Learning Model

ELM is a type of user or learner model that stores knowledge about the user (learner) in terms of a collection of episodes. In the sense of case-based learning, such episodes can be viewed as cases (Riesbeck & Schank, 1989). To construct the learner model, the code produced by a learner is analyzed in terms of the domain knowledge on the one side and a task description on the other side. This cognitive diagnosis results in a derivation tree of concepts and rules the learner might have used to solve the problem. These concepts and rules are instantiations of units from the knowledge base. The episodic learner model is made up of these instantiations and later generalizations based on them. To understand this form of episodic learner model, a short description of the knowledge representation and the diagnostic process will be given.

When using the environment, students code function definitions in a structured LISP-editor (Köhne & Weber, 1987), so their function code is at least syntactically correct. The cognitive analysis of the program code employs an explanation-based generalization (EBG) method (Mitchell, Keller, & Kedar-Cabelli, 1986). It starts with a task description related to higher concepts, plans, and schemata in the knowledge base. For every concept, a set of rules is stated describing different ways to solve the goal given by the plan of the concept. These rules are comparable to the notation of implementation methods for goals in PROUST (Johnson, 1986). There are "good", "suboptimal", and "buggy" rules explaining correct, correct but suboptimal, and incorrect solutions of the current goal or subgoal. The set of buggy rules is comparable to an error library in other ITSs (e.g., in the CMU-LISP-tutor (Anderson & Reiser, 1985). Applying a rule results in comparing the plan description to the corresponding part of the student's code. In the plan description, further concepts may be addressed. The cognitive diagnosis is called recursively until a function name, a parameter, or a constant is matched.

The cognitive diagnosis results in a derivation tree (Weber et al., 1993) built from all concepts and rules identified to explain the student's solution. This derivation tree is an explanation structure in the sense of EBG and is the basis for building up the episodic learner model. Concepts and rules addressed in the derivation tree are the basis for creating episodic frames. These frames are integrated into the knowledge base as instances of their concepts and rules. Slots in these instances refer to the context in which they were used (especially the current task), to the type of transformations of concepts, to the observed rules, and to the argument bindings.

In the next step, episodic frames are generalized. If an episodic frame is the first instance under a concept of the knowledge base, this single case is generalized from structural and semantic aspects in the data. This generalization mechanism is comparable to the single-case generalization in the EBG approach (Mitchell et al., 1986). Additionally, similarity-based generalizations of data and plans can occur. Generalizations represent the class of types of plans and corresponding data which will be used in further cognitive analyses to interpret the student's code. With increasing knowledge about a particular learner, hierarchies of generalizations and instances are built under the concepts and rules of the knowledge base. They constitute the episodic learner model.

The ELM-approach differs from the case-based approach in SCENT (McCalla & Greer, 1933) in several aspects. In SCENT, pre-analyzed cases are stored as a whole with respect to a static granularity hierarchy that expresses aggregation and abstraction dimensions. These cases are used during analysis of the student's code to give detailed advice. In ELM only examples from the course materials are pre-analyzed and the resulting explanation structures are stored in the individual case-based learner model. Elements from the explanation structures are stored with respect to their corresponding concepts from the domain knowledge base, so cases are distributed in terms of instances of concepts. Generalization hierarchies of instances are built up from explanations of the program code that a student produced to solve programming problems. Therefore, generalization hierarchies reflect the process of knowledge acquisition for a particular student. These individual cases – or parts of them – can be used for two different purposes. On the one hand, episodic instances can be used during further analyses as shortcuts if the actual code and plan match corresponding patterns in episodic instances. On the other hand, cases can be used by an analogical component to show up similar examples and problems for reminding purposes (Weber, 1991).

Empirical Investigations

Programming Behavior

Students learning a programming language will be differently successful when finishing the introductory programming course, so we explored how students performing "well" vs. "poorly" on the programming tasks of the final exam after the recursion lessons differently used the help facilities of the programming environment and how they differed in their programming behavior. Good students in the practical test spent more time planning and debugging their solutions in the first lessons (Tab. 1). From the beginning they tested and evaluated their programs more frequently and used the help environment more often. Especially the cognitive diagnosis, but also the stepper and trace tools were called more frequently than by less successful students. This behavior is not restricted to the debugging phase of programming; good programmers actively seek additional information about their program using the help system while still planning their first solution. Looking at the two final lessons about recursion, one can see that this behavior changes to the opposite. Now, better students actually spend less time planning and debugging. They also do not use the help system as much as before. These effects also hold for the programming tasks on the final exam, where recursive programming exercises had to be solved. To explain these results, one must consider solution quality. Students who become good LISP programmers use the visualization of the evaluation of LISP expressions more often during most lessons and they seem to explore much more in the first phase of learning to program. These findings are in concordance with the interpretation of the self-explanation effect (Chi, Bassok, Lewis, Reimann, & Glaser, 1989). Often these students already have a correct solution but actively vary and test it a lot. This makes them gain more experience with different types of exercises on which they can rely later. In the lessons about recursion (lessons 4 - 6) where exercises tend to be more analogous they switch strategies and purposefully use their knowledge to program a correct solution quickly. At this stage of the course solutions are less frequently correct at the first try, but if they are correct, students are certain of understanding the problem and tackle the next exercise without further testing and exploring the solution as they did in the first lessons.

Comparing Programming Environments With Different Support

During development of ELM-PE we observed a substantial change concerning students' success in the final programming test. The more support is available during the learning phase, the better students manage to solve the final programming problems when they are cut off from the advanced help system. We want to illustrate this by comparing the results from students working in ELM-PE with the results from previous courses. In a first course (Others Group), no advanced help and no visualization was available and students were only supported by a structure LISP editor or a simple text editor matching parentheses. In a second course (TALUS Group), no cognitive diagnosis was available, but the complete program code could be analyzed by our version of the

Table 1
Tendencies of Differences in Means for Students Performing "well" (N = 10) vs. "poorly" (N = 8) in the Programming Tasks of the Final Exam.

	Lesson						
	1	2	3	4	5	6	Final tasks
Planning time	+	+	+	+	0	-	-
Debugging time	+	+	+	-	-	-	-
Calling help texts	+	+	+	+	-	-	-
Use of STEPPER	-	+	0	+	+	+	-
Use of TRACE	0	+	0	+	-	+	+
Calling cognitive diagnosis	0	+	+	+	0	+	not available

Note: + "good" students use/need more/longer
- "good" students use/need less/shorter
0 both groups same

TALUS debugger (Murray, 1988). In the final test two tasks were the same for all courses. One, "list-until-atom", requires a terminating case that was not practiced in the course material and as such is unknown to students. The other, "count-item", is of the car-cdr-recursion type and was very similar to problems in lesson 6. Using ELM-PE, most subjects worked on both problems and more than 75% succeeded in solving the tasks (Tab. 2). The task "count-item" was solved by 86.7 % and 80 % of all students from the ELM-PE and the TALUS Group, respectively. But, only 45 % of the students from the Others Group who tried to solve this problem were successful. The problem "list-until-atom" was hardest to solve in all courses. While 76.9% of all students from the ELM-PE Group who worked on this problem solved the task correctly, in the TALUS Group 61.5 % and in the Others Group only 16% were finally successful. Generally, the courses with the most support available to students on request have been the most successful in the final practical exam where support is temporarily removed from students.

Discussion

In several developmental steps, ELM-PE changed from a structure editor to an elaborated programming environment supporting beginners learning to program in LISP. As the system has not included an explicit tutorial component up to now, it is not a complete ITS when compared, for example, to Anderson's LISP-tutor (Anderson & Reiser, 1985; Anderson et al., 1989), but more a knowledge-based help system. Students are allowed to choose the number and sequence of exercises on their own, though it is recommended to work out all exercises in the given order. The system interrupts students only if serious errors (e.g., syntax errors and errors using the interface) happen. Students are allowed to produce programming errors, the intention being that students learn from debugging and from reflecting on their bugs and misconceptions. Therefore, debugging and helping facilities specifically designed for beginners are integrated into the programming environment. The system remains passive as long as possible and offers advanced help only on demand.

Results from empirical studies with ELM-PE support this point of view. Though there was less support from human tutors during exercises, we observed fewer drop-outs and better performance in our last version compared to previous versions. Students who perform better in a final programming test use the programming environment for planning and debugging in their first phase of learning LISP. Often, after having produced a first correct result to a programming problem, they explore alternative solutions and try to understand how the LISP interpreter evaluates function calls by using the stepper. Students profit from their freedom to explore the system, to debug programs, to ask the system for help, and to follow the LISP interpreter evaluating LISP expressions. This result fits in with the active learning view. Good programmers seem to learn LISP through exploring the environment early and connecting each relevant information to the programming situation at hand. Green & Gilhooly (1990) have named this "problem finding", the advantage being that when performance is measured without additional help (in the final test), they can rely on well-learned programming strategies. As Chi et al. (1989) observed, practice as a factor in learning success seems to be greatly influenced by individual

Table 2
Percentage of Correctly Solved Programming Tasks in the Final Test for Different Programming Environments.

Programming Environment	Task	
	Count-Item	List-Until-Atom
ELM-PE	86.7 % ($n = 30$)	76.9 % ($n = 26$)
TALUS-PE	80.0 % ($n = 15$)	61.5 % ($n = 13$)
Others	45.0 % ($n = 20$)	12.0 % ($n = 25$)

learning strategies. Self-monitoring of one's own programming is very important. Accordingly, a programming environment should enable students to organize their programming in their own fashion.

As a second result, the evaluation study indicates that AI-techniques as they are used in the TALUS-analysis in the previous version of the programming environment and especially in the cognitive diagnosis of ELM-PE enhance the success of a programming environment for beginners.

References

Anderson, J. R., Conrad, F. G., & Corbett, A. T. (1989). Skill acquisition and the LISP tutor. *Cognitive Science, 13*, 467-505.

Anderson, J. R., & Reiser, B. J. (1985). The LISP tutor. *Byte, 10*(4), 159-175.

Bonar, J., & Cunningham, R. (1988). BRIDGE: An intelligent tutor for thinking about programming. In J. Self (Ed.), *Artificial intelligence and human learning: Intelligent computer-aided instruction* (pp. 391-409). London: Chapman & Hall.

Chi, M. T. H., Bassok, M., Lewis, M., Reimann, P., & Glaser, R. (1989). Self-explanations: how students study and use examples in learning to solve problems. *Cognitive Science, 13*, 145-182.

Green, A. J. K., & Gilhooly, K. J. (1990). Individual differences and effective learning procedures: the case of statistical computing. *International Journal of Man-Machine Studies, 33*, 97-119.

Johnson, L. W. (1986). *Intention-based diagnosis of novice programming errors*. London: Pitman.

Köhne, A., & Weber, G. (1987). STRUEDI: a LISP-structure editor for novice programmers. In H. J. Bullinger, & B. Schackel (Eds.), *Human-Computer Interaction INTERACT '87* (pp. 125-129). Amsterdam: North-Holland.

McCalla, G. I., & Greer, J. E. (1993). Two and one-half approaches to helping novices learn recursion. In E. Lemut, B. du Boulay, & G. Dettori (Eds.), *Cognitive models and intelligent environments for learning programming* (pp. 185-197). Berlin: Springer-Verlag.

Mitchell, T. M., Keller, R. M., & Kedar-Cabelli, S. T. (1986). Explanation-based generalization: a unifying view. *Machine Learning, 1*, 47-80.

Möbus, C., & Schröder, O. (1993). The acquisition of functional planning and programming knowledge: Diagnosis, modeling, and user-adapted help. In G. Strube, & K. F. Wender (Eds.), *The cognitive psychology of knowledge. The German Wissenspsychologie project.* (pp. 233-262). Amsterdam: North-Holland.

Murray, W. R. (1988). *Automatic program debugging for intelligent tutoring systems*. London: Pitman.

Neal, L. R. (1989). A system for example-based learning. In K. Bice, & C. Lewis (Eds.), *Proceedings of Human Factors in Computing Systems, CHI'89*, (pp. 63-68). Reading, MA, Addison-Wesley.

Riesbeck, C. K., & Schank, R. C. (1989). *Inside case-based reasoning*. Hillsdale, NJ: Lawrence Erlbaum.

Weber, G. (1991). Explanation-based retrieval in a case-based learning model. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, (pp. 522-527). Chicago, IL.

Weber, G., Bögelsack, A., & Wender, K. F. (1993). When can individual student models be useful? In G. Strube, & K. F. Wender (Eds.), *The cognitive psychology of knowledge. The German Wissenspsychologie project*. (pp. 263-284). Amsterdam: North-Holland.

Acknowledgements

This work was supported by the German Science Foundation under Grant We 498/12.

Interactive MultiMedia Distance Learning (IMDL)
The Prototype of the Virtual Classroom

JACK M. WILSON PH.D., DIRECTOR, DAVID N. MOSHER, PROJECT MANAGER
Anderson Center for Innovation in Undergraduate Education (CIUE)
Rensselaer Polytechnic Institute
Troy, New York 12180
E-MAIL: Bitnet JWILSON@RPITSMTS

Abstract: This paper describes a collaborative project between Rensselaer's CIUE and AT&T's Bell Laboratories to create a prototype of an interactive, distributed multimedia desktop classroom environment. The prototype is called a "Virtual Classroom" because it brings instructors and students from remote sites together in a desktop computer environment that includes multi-point video, audio and multimedia data communication. This environment fosters real interactivity between students and instructors and among students who may be located at sites anywhere in the world.

Introduction

Interactive Multimedia Distance Learning (IMDL) is the model for the "Virtual Classroom" prototyped by Rensselaer and AT&T. The IMDL environment model combines the rich communication capabilities of 2-way video teleconferencing with real time, synchronous data communications for sharing of computer generated examples and data. The level of interaction is quite high and the environment offers the possibility of sharing between teacher and students and students with their peers.

The visual and auditory communication is enabled by multi-point video conferencing over the AT&T Digital Network, using Integrated Services Digital Network (ISDN) Primary Rate Interface (PRI) facilities. The 2 way video communication is integrated into the desktop computer environment via a video window on the computer screen. Instructors, and students, may control MS Windows-based applications on the other participants' workstations. The shared applications may include instructional applications, text and graphics screens, animations, video clips and audio clips used to enhance learning and collaboration.

CIUE is a research and development center at Rensselaer dedicated to the advancement of innovative undergraduate education. The CIUE is particularly interested in the applications of technology and cognitive science in education. We also act in a leadership role for the implementation of research in the classrooms of our campus and other institutions internationally.

AT&T Bell Laboratories is the Research and Development Center for AT&T and is an international leader in research in telecommunications and computing.

Learning in the Virtual Classroom

Let's imagine for a moment that you are a mother and a returning student in an engineering school. You are taking classes at home because you have family responsibilities. You have spent the morning doing library research at your home computer and now it's time for Physics class......

You save your journal articles on your local disk drive and then click on an icon on your screen to sign on for your class. After entering your name and ID information, a video window appears on your screen where you see the instructor giving announcements about today's class. "Just in time," you exclaim, as you enlarge the video window and turn up the volume in order to see and hear better. The instructor then begins to greet each student individually and switches the focus of the session to each location so that everyone may see and hear each student saying hello. You recognize other students who are participating in this class from home and on-campus locations.

The class begins with the instructor discussing the concept of acceleration. The instructor then explains a demonstration of gravitational acceleration that she is about to perform. You decide that this looks important and turn on your video capturing function in order to view the demonstration again at a later time. After performing the demonstration, the instructor then sends each student a copy of the demonstration, which is a digitized video clip of an assistant dropping a ball. Now the instructor begins to guide you through the process of graphing the position of the ball, stepping through the video clip frame by frame; writing notes on your screen to point out important points.

One of the steps of the exercise makes absolutely no sense to you and you send the instructor a question message. Apparently, you are not the only student confused. The instructor stops the session and acknowledges that she has received a lot of questions all of a sudden. The instructor then turns the focus of the class to you and you ask your question and point out on the screen where you got confused After the instructor clears up the point of confusion, you add some notes to the screen for later use. The instructor then poses several problems for the class to solve on their own, using the data collected from the graphing exercise and the computational tools that you use every day for all your classes. After you perform the required analyses, you send the file to the instructor for review. The instructor then displays each student's work and writes comments about their methods.... .

This scenario may sound futuristic; however, it is based on the tools and capabilities of the IMDL prototype we have developed. The actions described in the scenario are detailed further below to show how they enable an efficient learning environment. The technical capabilities required for this kind of interaction to occur at home locations are rapidly being deployed by telephone and cable companies. Refer to the section entitled, "How Will Technological Advances Effect IMDL?" for further discussion of this issue.

We are attempting to recreate the "social construct" of the classroom. In order to recreate the traditional classroom, we must offer the capability to the participants to perform the actions that they perform in traditional classrooms. We will not attempt to discuss all the teaching strategies that educators have used, or how they may be implemented in the IMDL environment. However, in the next section we will discuss a few of the actions common to most teaching strategies and how we have implemented them in the IMDL environment

Interactions in the Virtual Classroom

In traditional classrooms, teachers may create exercises for students to work with in a shared space (the blackboard). Teachers may also write comments on student's work. In addition, teachers use text, graphics, animations, videos, sounds or live demonstrations to present curriculum content to students. Students may work with exercises from the teacher in local (on desktop) or shared space (the blackboard). They may also collaborate on exercises with other students. Also, students may alert the teacher that they would like to ask a question.

Event Sharing

In the IMDL environment teachers and students share a workspace of identical instructional software applications. One participants' workstation is configured to be the leader of the session and all the other workstations follow the actions of the leader. The actions may be to present screens of text or graphics, play animations, video or audio clips, or simulate a phenomenon. The leader may pass the focus to a student workstation. When this happens, the student controls the flow of the shared events. The focus for video and audio may also be passed to the student, so that everyone may see and hear the student ask their question. However, in our model only the instructor may pass the focus of the session and only the instructor may take it away. This instructor control is important in order to maintain class structure in the distributed classroom.

Each workstation may work independently, as well, both during the session and on their own time, with all the instructional software applications. The students may want to research something that has just been discussed in class, or may need to perform a calculation to answer a question posed in class. These actions may be performed with local software applications.

Annotation

The participant whose workstation is designated the leader may draw on their own screen. These drawings will also appear on the other participants' screens. This feature of IMDL allows the leader to add notations to the material being presented, adding meaning and emphasizing the important points.

Hand Raising

The students may send a message to the instructor's workstation, letting the instructor know that they have a question. The instructor will see the message coming in and may choose to acknowledge the question or to respond at a more convenient time. When the instructor acknowledges the student, then the instructor may pass the focus of the session to the student (as we discussed in the section above on Event Sharing). Our testing has shown that this structured approach to teacher/student dialogs is helpful because it helps each participant to be sure who has the focus and who is sharing information with the other participants.

Instructor Pointer

The teacher may move a pointer around the student screens to direct their attention to the important points in the information displays. The teacher does not need to verbally describe the position of the point of interest. This shared pointer works similarly to the pointers used in traditional classrooms. The students also have their own pointers (the MS Windows mouse cursor) that they have control over, independent of the teacher pointer.

Implications of Learning Theory

At the core of any discussion of education is the fact that education is a communication process. Regardless of the model of pedagogy that one works with; communication between instructor and student and among students is central to the process. New developments in theories of learning make communications even more important to the process. We bring to this project the benefit of a great deal of education research. What this modern research is telling us may be summarized by the following points:

Teachers are not simply the delivery mechanisms of the content of a curriculum. A great lecturer can be very motivational, but research has demonstrated that the lecture is not efficient in stimulating student learning. (Laws,1991; Hestenes,et al,1992; Redish,Wilson, McDaniel,1992) The model we work with is one of a teacher as "coach" of their students' learning process (Pea,1992; Laws,1991). The teacher is actively involved in helping the students construct their own understanding of the curriculum material. This activity is best carried out when the teacher and students have available to them a rich set of communication tools to enable the coaching process.

Learning is a highly interactive process. Students bring with them a highly developed set of preconceptions and beliefs. The teacher and student become involved in a learning "conversation" in which both parties clarify messages, test for understanding, compare and contrast with previous understandings and are both transformed by the experience Pea (1992). The learning "conversation" is a communication process in which meanings are negotiated to student understanding of curriculum material. What we seek to provide are a rich set of communication tools to help foster these interactions between teachers and students.

Education has been criticized as an endeavor that is too far removed from the context of its meaning. If learning is to be viewed as a process that has meaning beyond the classroom; the students must be able to reach beyond the classroom. Practitioners from the field of study can be brought to the classroom. However, this is not always possible, or practical. There have been several excellent examples of providing electronic mail and conferencing between students and practitioners. The "National Geographic Service Kids' Network", Julyan (1991), allowed students to collaborate with other students and scientists on a study of acid rain. The "Collaborative Visualization Project", Pea (1992) , a new project now being implemented, will allow students to collaborate with other students and access on-line data resources and scientists in a Project Enhanced Science Learning-at-a-distance approach. These projects each are important steps in bringing the student into the communities of practitioners.

Learning can be enhanced by providing students with access to powerful computing tools that can allow the student to interact with real data and solve open-ended problems. Learning-by-doing has been shown to be a successful pedagogical model to enable students to solve real world problems. (Laws, 1991; Redish, Wilson, McDaniel,1992) This learner-centered view allows the student to start with what they know and build their own understanding of the subject. This approach also has the advantage of supporting individual differences in learning styles. Students bring to the classroom a diversity of interests, levels of preparation, cultural backgrounds and learning styles. Included in our model is a set of powerful software tools that allow data to be captured. abstract concepts to be simulated and sets of data to be visualized. We advocate placing these tools for learning in the hands of each student, not only in the hands of the instructor for recitation.

Cooperative learning is a highly structured, systematic instructional strategy in which students work in small groups toward a common goal. This strategy has been shown to promote active learning, positive student attitudes toward learning and increase student interdependence. Increased interdependence is a positive goal for students because of its effects on student interpersonal skills, teamwork capability and self esteem. While working in teams on a project, students cannot be passive on-lookers; the contribution of each team member is important. (Millis. 1991) Teamwork is also becoming a widely implemented organizational strategy in many work settings, including manufacturing, services and government. Instructional practices should prepare students for working in this type of environment. Our IMDL model allows the instructor to set up groups of students who will work together on an assignment and then share their results with the group.

Comparison of IMDL Environment with Existing Distance Learning Environments

Traditional distance learning programs have most often replicated the teacher lecture model of education These environments primarily utilize one-way satellite transmission of video signals, with limited student feedback via phone calls. Interaction between the teacher and students is limited and interaction between students at different sites is non-existent. The technology is not at fault; the problem lies with the types of activities to which it is being applied. This technology may be appropriate for delivering messages to multiple sites when interaction is not desired. However, it lacks any significant interaction that might foster teacher "coaching" of students, or learning conversations between teachers and students or students and their peers.

Other technologies used in distance education offer more interaction but support either video and audio interaction, or data interaction, but not both modes. Each mode has unique characteristics that allow enhancement of communication but suffer from limitations also.

Video teleconferencing offers 2 way and multi-point video and audio communication between sites. Video's strength lies in its ability to closely mimic "reality." Face to face communication is full of subtle actions and nuances that modify the meaning of the spoken words. Students also may see examples and demonstrations being performed. Unlike a "real" classroom, though, students cannot go to the blackboard and interact with the teacher's example or view other students' work.

Computer conferencing is primarily a means for sharing text or graphic messages that are deferred in time This allows the sharing of work between teacher and students and students with their peers. Verbal communication is transmitted in text form, also deferred in time. This environment lacks the rich communication capability of video, because of the communication limitations of text and graphics. Sharing of a common data set for collaborative work is also rare in this environment, although the NGS KidsNet, Julyan (1991), does allow students to work with common, aggregated data to interpret acid rain studies.

The IMDL environment model that we have developed combines the rich communication capabilities of 2-way video teleconferencing with real time, synchronous data communications for sharing of computer application "events" between workstations. The level of interaction is quite high and the environment offers the possibility of sharing between teacher and students and students with their peers. The interactions are real-time, not delayed in time. This capability increases student motivation and may provide increased interaction and feedback between students.

Our learning model is a student-centered model; therefore, students and teachers all have the same software applications and databases installed in their desktop workstations. Students may work on their own with the software tools or use on-line tools as references while participating in a class session.

Teachers may use the IMDL environment for lecture when appropriate, or to guide students through exercises performed on their local workstations. Students may share their work with all the other participants and receive feedback. They may also engage students in learning conversations that include passing the control of the session back and forth from teacher to student.

Project Background

The Interactive Multimedia Distance Learning project evolved from discussions between RPI and AT&T about how to use the strengths of AT&T's Bell Labs and RPI's CIUE to collaborate on a project in the domain of computers and communications. AT&T Bell Labs planned to contribute their expertise in communications and networks and CIUE planned to contribute experience in developing multimedia instructional software and managing software development projects, such as CUPLE, the Comprehensive Physics Learning Environment (Wilson, Redish,1992). The CUPLE project integrates hypermedia based computer activities and video material on the same computer screen. The environment also allows for live data acquisition from laboratory interfaces and from video sources. CUPLE has been created by a consortium of universities and is now in the testing stage at over 200 universities. Physics Academic Software, the publishing arm of the American Institute of Physics, is publishing the final version.

From these discussions it was decided that the project would involve re-designing a course from AT&T 's University of Sales Excellence (USE).. USE is an internal training and education organization supporting marketing staff in the Business Communication Services division of AT&T. The course chosen for redesign is titled, "How to Make Money Grow on Trees." This course teaches the features of AT&T Advanced 800 Services and how to apply them to customer applications.

Currently, the course is delivered in audio graphic mode. The students attend the course in a training room at their office site equipped with a conference speakerphone, with a large microphone, electronic polling devices and a video blackboard. The students communicate verbally with the instructor using the speaker phone and the AT&T Alliance Call Bridging Service in a multi-point voice conference. All the student sites hear the instructor's lecture, and the student questions from other sites, on their conference phone speaker. The students follow the instructor through the course content material in their Student Guides. The video blackboard is used to display text and graphics screens, with instructor annotation, to supplement the Student Guide materials. Also, students may give answers to multiple choice questions by pressing a button on their electronic polling devices.

We wanted to apply the IMDL environment model to a real life situation. Using our model, we would make the course environment more interactive. In the old version of the course, students could not see their instructor or the other students. They also had no way to share their work with the instructor or other students. The instructional strategy was primarily lecture and the students were only passively engaged in the experience. The IMDL environment model of instruction would also allow students to construct their own understanding of the material by giving them powerful on-line software tools and reference materials that they may access during the course.

We also wanted to investigate how the interactions between teacher and student and interactions between student and their peers change when the learning environment changes from a lecture environment to a highly interactive desktop computer based video conference environment.

Initial Findings from Prototype Testing

In June of 1993, we tested the IMDL environment with live participants. The University of Sales Excellence course that we developed was delivered by an instructor in Cincinnati, Ohio. The student participants were located in Dallas, Texas, Chicago, Illinois and Holmdel, New Jersey. The "students" were actually other AT&T instructors, who were volunteering to participate. This was done for 2 reasons. First, we did not want real student's success in the course to be impeded in any way by any unforeseen problems that developed in the delivery of the course. Second, instructors are accustomed to thinking about teacher-student interactions and we felt that this would give us better feedback.

The participants' reactions might be grouped into 2 areas. The first group of comments were positive about the user interface and software tools. Generally, the students enjoyed the experience. They also found that it had great potential to improve upon other modes of instruction delivery. The second group of comments described the participants' intimidation by the technology employed in delivering the course. The students and instructor felt that they would need some help in getting started using IMDL technology to deliver instruction. This reaction is understandable and may be addressed in the design of the IMDL interface and training materials.

How Will Technological Advances Effect IMDL?

The only thing that is certain is change itself. This couldn't be more true than in the field of educational technology. Our IMDL model is designed to use off-the-shelf technology and to accommodate technological change. We want to be able to port our model to whatever platform is accepted as a standard in the future.

The pace of technological change in the telecommunications industry is such that the cost barriers to entry are rapidly falling. The costs of the high bandwidth transmission necessary for IMDL may be prohibitive for some organizations and it is certainly difficult to justify for the home. Cable companies and phone companies are competing and collaborating to make high bandwidth transmission universally accessible. This will make implementation of IMDL environments easier to cost justify.

There are other technical hurdles to overcome in the implementation of IMDL environments. These mainly concern better digitization of video and resource management of large libraries of multimedia material (video, graphics, etc.). Developments in these areas are also proceeding well and the implementation of multimedia servers and new video compression algorithms will make the technological hurdles easier to overcome in the near future.

Applications of IMDL

The 1990's are presenting many challenges to institutions of higher education. Private institutions are facing declining enrollments as the number of college age students decreases. They are also facing the need to lower costs as students are finding it increasingly difficult to afford private education. On the other hand, public higher education institutions will probably experience level or increasing enrollments due to many factors. Among these factors are returning adult learners, the shift to more affordable education choices, and geographic population shifts. In addition, all institutions will need to accommodate a more diverse student population; including adult learners, cultural diversity and workforce continuing education.

IMDL will help address these challenges in several ways. Institutions will be able to create a more productive learning environment, with more learning going on in a shorter amount of time. This will address cost issues, as students will be able to complete programs of study more quickly. Students will have access to more effective learning environments in a greater variety of locations. IMDL will enable instructors to reach out with course materials to locations outside the classroom and off the campus. This will address the needs of all institutions to reach out to a wider student body and enable life long learning for our entire population. In order for our educational institutions to provide more productive learning environments to a diverse student population institutions must share resources and collaborate in the delivery of instruction. IMDL will enable this kind of cooperation.

Finally, and certainly as important, are the needs of our secondary schools. They face a number of challenges for the 90's. Secondary schools have had more difficulty keeping up with technological advancements in industry and higher education. Teachers in secondary schools face isolation from their peers and the need to keep up with education technology. In addition, secondary schools face the familiar problems of lowering costs and providing a more productive learning environment with a lower budget. IMDL environments address these issues by offering the opportunity for teachers and students to reach beyond the classroom to receive instruction, collaborate/interact with peers in learning projects, receive faculty development programs and link to resources not even thought of today. Costs will be the major issue in creating these links, but the investments must be made.

Indications for Further Research

Our work to date has focused a "proof of concept" test of technology and techniques for IMDL. Further work needs to done to study the cognitive and pedagogical implications of learning in distributed classrooms. Our future work will concentrate on further testing IMDL these implications in the delivery of higher education courses, corporate training courses and faculty development workshops. We want to answer questions such as the following:

1. Does IMDL allow instructors to guide students to learning objectives requiring higher cognitive level thinking?

2. Compare the cost effectiveness of instructor led, Computer Assisted Instruction, non-interactive distance learning and IMDL instruction delivery.

3. What kind of cues will best alert users to system status during sessions?

4. What kind of preparation training will make the IMDL environment comfortable to use for the participants?

5. What kind of class activities are best performed locally and what activities are best performed in conference?

References

Hestenes, David, Malcolm Wells and Gregg Swackhamer. (1992) Force Concept Inventory. *The Physics Teacher.* Vol. 30 pp. 141-158.

Julyan, C. (1991, Spring) Getting Connected to Science. *Hands On.* (14)1, 4-7

Laws, Priscilla. (1991) Workshop Physics: Learning Physics by Doing It.
Change Magazine July/August 1991

Millis, B.J. (1991) Fulfilling the Promise of the "Seven Principles" Through Cooperative Learning: An Action Agenda for the University Classroom. *Journal on Excellence in College Teaching*, Vol. 2 pp. 139-144.

Pea, Roy and Louis M.Gomez. Distributed Multimedia Learning Environments: Why and How? (1992) *Interactive Learning Environments* Vol. 2 Issue 2 pp. 73-109

Redish, E.F., Jack M.Wilson and Chad McDaniel. (1992) The CUPLE Project: A Hyper- and Multimedia Approach to Restructuring Physics Education. In E. Barrett, ed., *Sociomedia-Multimedia, Hypermedia and the Social Construction of Knowledge* Cambridge, MA, MIT Press

Wilson, Jack M. and E.F.Redish, (1992) The Comprehensive Unified Physics Learning Environment: Part I. Background and System Operation, Part II. The Basis for Integrated Studies. *Computers in Physics* Vol. 6, No.2, and Vol.6, No.3

A User-Adaptive Interface for Computer Assisted Language Learning

EVE WILSON
Computing Laboratory, University of Kent at Canterbury
Canterbury, Kent, CT2 7NF, England
E-mail: ew@ukc.ac.uk

Abstract: To meet the needs of an individual student an adaptive CALL package must have a dynamic model of student performance, a means of varying the difficulty of the learning task, and a mapping between student competence and task complexity. This paper analyses how these components can be implemented for lexical, syntactical and discourse skills using domain knowledge from the Oxford Advanced Learner's Dictionary and the Susanne Corpus, a fully tagged subset of the Brown Corpus.

Rationale for Computer Assisted Language Learning

Students who are learning English as a Foreign Language in an English-speaking country form a multi-cultural group with disparate motivations and goals, whose diversity cannot be adequately accommodated in a traditional classroom. There is no rate of imparting information nor sequence of instruction that will meet the needs of every student. Teachers can only construct in their mind a model of the 'average' student and have as their goal the adequate progress of this hypothetical student; frequently they are guiltily aware that the more able will be bored while the less able will be lost. Today computerised language learning packages offer the majority of students their only realistic opportunity for individual tuition. Even with the most mundane of CALL packages, each student can essay an answer to every question, repeat lessons not fully comprehended or skip lessons that merely rehearse previously acquired skills. A CALL package which permits a student to make this sort of choice is sometimes referred to as a *user-tailored* system.

Self-directed learning of this kind has been shown to benefit *mature learners*, that is, experienced students of proven academic competence (Kearsley & Hillelsohn, 1982), who are capable of determining their own educational needs. However, less experienced or less able students may not be able to make an accurate assessment of their own shortcomings or to devise for themselves a coherent and comprehensive study plan. For these students, a CALL package can be immeasurably improved by a user-adaptive interface i.e. an interface where adaptations in the order and pace of learning are made by the system, not by the student.

User-Adaptive Interfaces

A *user-adaptive interface* is one which can change its behaviour automatically in response to its experience of user performance, that is, it changes to suit the skills and knowledge of an individual student. The system designer or language teacher no longer tries to construct an interface for a stereotypical average user, to which no real learner conforms. Instead, the designer accepts that no single learning theory pattern is suitable for all students – or even for one student over a range of skills or a period of time – and realises that the teaching package must adapt to the student's varying abilities in different skills by providing information and exercises at a level which matches the student's current needs. To this end an adaptive learning package must have three components:

- a means of recording and measuring individual student performance. This could also be described as a dynamic model of the user's past performance and current capability. This dimension of capability is inevitably continuous rather than discrete.

• a means of adjusting the learning task so as to change its difficulty. In the context of language learning this might simply be the ability to offer appropriate help and linguistic explanations to students of various levels of ability and to set for each student exercises commensurate with the student's expected performance.

• mappings of help level onto student capability and of student capability on to degree of difficulty. How these mappings are achieved determines the type of user-adaptive interface.

Types of User-Adaptive Interfaces

There are two main types of user-adaptive interfaces for language learning: discrete-step interfaces and continuously variable interfaces.

Discrete-step interfaces

A *discrete step interface* is an interface which identifies the user as a member of a particular ability group and sets the interface to correspond with the skill level for the group. This, while more satisfactory than a general user interface, does not allow for continuous variation in student capability but assumes discrete ability levels. In language learning it is a practical solution for the organisation of *expert procedural knowledge*, which is the distillation of the rules that govern the lexical, syntactical, and discourse structures of the language. The traditional presentation format for this knowledge has been textbooks and reference books. Such books might be aimed at a specific category of language learner e.g. *novice*, *intermediate*, *expert*, or might be an exhaustive exposition of all facets of a language for scholars of comparative linguistics and languages. Knowledge of this kind has never formed a continuum: each text represents the independent view of one teacher or expert on the information needed by an *average* learner at a *typical* stage in the learning process. The order in which the information is presented may be said to represent the author's strategy for how best to accomplish the learning task. Computerised learning packages, particularly those implemented in hypertext, allow us to separate monolithic texts into fragments and to convert these fragments into many different virtual structures. This is useful for accommodating students whose ability over different linguistic skills varies widely: in one skill a student may be given a simple explanation from first principles while in another the same student may be given only a brief hint or reminder because past performance indicates a consistently high level of ability. In a skill where the student's ability is intermediate, extra information and/or explanation of more abstruse points may be provided. But, however skilful the dissection of the information and however varied the virtual knowledge structures in which it can be stored, it is not possible for the computer, to re-write or even re-phrase the expert knowledge to cater uniquely for the needs of an individual student as a human teacher might be able to do. Thus, the interface for expert procedural knowledge remains a discrete-step interface.

Continuously variable interface

Although levels of procedural knowledge are discrete the domain knowledge in any natural language is, of course, a continuum. *Domain knowledge* consists of the accumulated oral and written records that are accepted by the native speakers of this language. While the documents that comprise these sources may in themselves be as discrete as the expositions of procedural knowledge, together they form a continuum that represents every aspect of language use in every sphere of human activity. That this is so, is recognised by the accumulation of corpora – collections of texts from different genre assembled to enable language scholars to investigate the evolution and current state of a language. The remainder of this paper will discuss how corpora may be used to build an interface which uniquely adapts to the linguistic needs of an individual student in the generation of exercises to improve the student's linguistic skills. As a first step, it will look at ways of recording and measuring student performance.

Recording and Measuring Student Performance

Before designers can build CALL systems that 'understand' their users, they must be able to analyse the interactions between the user and the computer in the language learning task. This means that they must be able to specify the skills that make up the learning task and must have ways of measuring student performance in individual skills. Language learning skills may be divided into three major skills categories:

Lexical skills appertain to words;
Syntactical skills are the skills needed to produce grammatically correct phrases or sentences.
Discourse skills are the skills needed to write coherent and cohesive text.

Obviously, all these skills overlap but nevertheless they provide the parameters for creating a student performance model The parameters are given values which indicate the student's past performance in the relevant skill. The student's attainment can be assessed simply by measuring the student success in linguistic exercises to assess and improve the skill. In this way, student competence can be viewed as a continuum. The student exercises should, of course, be set at a level which is appropriate to the student's current attainment so that every student is stretched to the limit of current ability: competent students should not be bored, weaker students should not be intimidated.

An initial user profile

The first task is to devise an initial user profile. This can be done simply by using a standard model for novice, intermediate, or expert. This model will rapidly self-adjust to give a more accurate assessment of proficiency in each skill as the student uses the package. A further refinement is to incorporate into the student model some element that reflects the average performance in a particular skill of students from different language groups.

Ensuring a Continuum of Exercises

The second task is to ensure a continuum of exercises. This is only possible if the CALL package has the ability to assign to an exercise a parameter which gives a measure of its expected difficulty. How this can be done depends upon the form of the domain knowledge on which the exercise is based. There are two main forms: continuous text and pre-selected passages, sentences or phrases that have been chosen by experts to illustrate precise linguistic features.

- Continuous text

A simple way to assign a coefficient of difficulty to continuous text is to use one (or more) of the methods that have been developed to assess readability grades; the more difficult a passage is to read, the more demanding should be any exercise based on it. Two common measures of readability are: Gunning Fog Index (Gunning, 1952) and Information Density (Wainwright, 1984)

Figure 1 shows Fog Indices and Information Densities for texts in the Susanne Corpus, (Sampson, 1992). The coefficients obtained from these two methods do not always correlate and other factors must also be taken into account when selecting material. However, either method gives a continuous scale of difficulty which can be correlated against student proficiency.

File	Snts	Wrds	Lwrds	Nouns	Adjs	Fog	ID
A01	88	1939	712	97	88	10.75	41.72
G01	86	2069	465	181	76	11.81	31.22
J01	65	1908	629	161	91	15.45	41.40
N01	156	2065	391	65	85	5.66	22.08

Figure 1. Gunning's Fog Index and Information Density coefficients for texts in the Susanne Corpus

- Pre-selected passages

A general text database is not always a suitable source for language exercises, particularly for advanced lexical exercises, e.g. differentiating frequently confused words such as *early, soon*. There are two reasons:

- there is no guarantee that the word has been correctly used;
- the database may not contain enough usage examples to create a worthwhile exercise.

A more appropriate source for lexical exercises is a dictionary. Here the usage coverage for every word is exhaustive and the quality of the examples is assured. However, because the examples consist almost entirely of single sentences or phrases, it is not possible to use Gunning's Fog Index or Information Density to assess difficulty. For exercises where there is an abundance of material (e.g. verb particles) other methods can be used to choose appropriate examples. One of these is based on word frequency counts, e.g. use only verbs that are in the most frequently used 1000 words, 1500 words, 2000 words and so on. The word counts may be determined either from general literature or from texts relating to the student's primary discipline: the latter will increase student motivation. (Sometimes, even in a dictionary, usage examples may be inadequate.)

Hence, the type of exercise and the preferred source for an exercise of that type both influence implementation algorithms.

Implementation of the Exercise Generation Package

Exercise generation is dynamic. Once the student has decided what linguistic skill to work on, there are four stages to the generation process:

Determine a suitable source text
Retrieve passages which illustrate the required linguistic feature
Sieve the retrieved examples to leave those examples most suited to the needs of the current user.
Generate the electronic version of the exercises.

Determining a suitable source

Initially, only two text sources, or knowledge domains, were readily available to this project. – the Oxford Advanced Learner's Dictionary (OALD) (Oxford, 1974) and the Susanne Corpus (Sampson, 1992). Consequently, it was decided simply to have a two-column table of skill against source to determine the more appropriate knowledge domain. If the source assigned is the Susanne Corpus (i.e. a continuous text source), the program uses the parameter for student competence in that skill to compute a commensurate readability grade and then selects the Susanne text that matches that grade most closely. As the number of sources available to the project increases, a more complex algorithm may be needed. A project priority is to investigate automatic parsing so that texts directly related to a student's prime discipline can be used by the exercise generator. This should increase student motivation by making exercises more directly relevant to student need.

Choosing suitable passages

Once a source has been selected, there are two main methods of retrieving appropriate examples: direct lexical search and searching for tags.

- **Direct lexical search**

This is the more straightforward search and needs little explanation. The program searches the selected source texts for occurrences of a precise string of letters, usually a word, occasionally a word-stem. The technique can be used with tagged or untagged sources. An indexed text affords a considerable increase in speed of retrieval.

- **Searching for tags**

Complicating use of tagged sources is that every major database uses a different tagging system. Hence, the linguistic parameters for retrieval and even the retrieval algorithm vary with source. Consequently, rather than explain algorithms in detail, it might be more generally useful to compare the two main tagging systems encountered in this project and highlight features of each that are particularly valuable in exercise selection.

- Susanne Corpus

The Susanne Corpus is a 128,000 word subset of the Brown Corpus (Francis, 1989) comprising 64 files, each of more than 2000 words from four Brown genres:

A	press reportage	J	learned (mainly scientific and technical writing)
G	Belle lettres, biography, memoirs	N	adventure and Western fiction

The texts in Susanne have been manually analysed and annotated in a way that gives access to both surface and logical structure. Figure 2 shows a portion of text from Susanne. The most valuable fields in Susanne for exercise extraction are the *word* and *lemma* fields for finding examples for lexical exercises; the *wordtag* field for finding occurrences of syntactical features and for finding discourse frameworks and, occasionally, the *word* field for lexical queries. The *parse* field has not yet proved useful for selecting discourse passages, but structural complexity may give another measure of degree of difficulty of a text.

Reference	Status	Wordtag	Word	Lemma	Parse
G04:0010i	-	CC	and	and	[S+.
G04:0010j	-	AT	the	the	[Ns:s.
G04:0010k	-	NN1n	game	game	.Ns:s]
G04:0010m	-	VVDv	ended	end	[Vd.Vd]S+]S]
G04:0010n	-	YF	+.	-	.

Figure 2. Tagged text from the Susanne Corpus

- Oxford Advanced Learner's Dictionary

In contrast with the Susanne Corpus, the example sentences and phrases in the OALD are not parsed. Thus, it is sometimes difficult to use these examples for the automatic generation of syntactical exercises. Tagging in the OALD is primarily associated with the headword for each dictionary entry, or, where a headword has several senses, with each sense. Grammatical information is detailed and includes plurals, comparatives and superlatives. Idiomatic expressions and phrasal verbs are fully enumerated and well defined. Words and expressions that fall into specialist English registers (eg accounts, aerospace, algebra, etc) are labelled. Every verb is classified by how it can be used, e.g.

[VP18A] S + vt + noun/pronoun + infinitive

I felt the house shake

Many of the verbs can take several patterns and these are listed together after the sense number: e.g.

) = 6A, 8, 9, 10, 18A, 19A, 24.

then follows the definition and then the example sentences. Unfortunately there is not an explicit link between a verb pattern and the instantiation of that pattern. Consequently, it is difficult to extract an illustrative sentence automatically.

Sieving the retrieved examples

Retrieved sentences are sieved for two purposes:

- to remove as far as possible all sentences that may lend themselves to more than one interpretation, e.g. in an exercise on pronouns it should be quite clear from the context which pronoun to use.
- to retrieve only enough examples to create an adequate exercise. Selecting sentences at random from those retrieved ensures that students will be given a different exercise every time they rehearse that linguistic skill; however, if the exercise has not already been graded for difficulty, the use of word frequency counts is preferable.

Generating the exercises

Where appropriate the retrieved examples are re-ordered. Most exercises are generated as Cloze exercises that can be completed by the student using only a pointing device, e.g. a mouse. This has the disadvantage that it only tests recognition, not recall, but it requires little manual dexterity so the student can do it quickly. The hypertext templates used are described in (Wilson, 1992). Figures 3 shows exercises on *pronouns* generated from the Susanne Corpus; figure 4 shows a specialist vocabulary exercise for a student of architecture generated from the OALD

Complete the following sentences by choosing the correct pronouns:

he	**him**	**himself**	**she**	**her**	**herself**
it	**it**	**itself**	**they**	**them**	**themselves**

They saw it before ___ did, even with my binoculars.
Occasionally, for no reason that I could see, ___ would suddenly alter the angle of their trot.
For ten minutes ___ ran beneath the squall, raising their arms and, for the first time, shouting and capering.
___ bent down, a black cranelike figure, and put his mouth to the ground.

Figure 3. Part of an exercise on pronouns derived from Susanne Corpus, G04, readability grade: 7.31

Select an architectural term from the column on the left.
Then, from the column on the right, select the definition that corresponds.

caryatid	*(either end of the)transverse part of a cross-shaped church*
coping	*curved edge where two vaults meet (in a roof)*
corbel	*draped statue of a female figure used as a support (eg a pillar) in a building*
Corinthian	*high, narrow, pointed arch or window*
cornice	*line of (sometimes overhanging) stonework or brickwork on top of a wall*
Doric	*column in ancient Greek architecture, with a decoration of leaves on the capital*

Figure 4. Part of a specialist vocabulary exercise on architecture from OALD

Evaluation

The format of the exercise is popular: students have no difficulty in using it and are generally enthusiastic. The feedback given to them is clear and they have no problem in monitoring their progress. What is not clear from the evaluation so far is to what degree there is a correlation between readability grade and exercise difficulty. For certain linguistic skills, such as use of articles, students' ability in the skill seems not to vary with readability grade of the text. This may not matter, but it requires further investigation. Other exercises, which were difficult to classify initially, such as differentiating word pairs, *(early, soon) (imply, infer) (constantly, continually) (anticipate, expect)* seem to be uniformly difficult even for native speakers. However, testing is incomplete; in particular, tests involving students with more widely differing levels of ability are a priority.

References

Francis, W. N., & Kucera, H. (1989). *Manual of Information to accompany a Standard Corpus of Present-Day Edited American English*, Department of Linguistics, Brown University, Providence, Rhode Island.

Garside, R. G., et al. (1987). *The Computational Analysis of English*, Longman, London.

Gunning, R. (1952). *The Techniques of Clear Writing*, McGraw Hill, New York.

Kearsley, G. P., & Hillelsohn, M. J. (1982). "Human factors considerations for computer-based training", *Journal of Computer-based Instruction*, 8(4), pp. 74-84.

Oxford University Press (1974). *Oxford Advanced Learner's Dictionary of Current English*, Third Edition, Edited by A. S. Hornby, OUP, Oxford.

Sampson, G. R. (1992). *The Susanne Corpus*, University of Sussex.

Wainwright, G. R. (1984). *Report Writing Management Update*.

Wilson, E., Lindley, L., & Pullen, S. (1992). CALLGUIDE: using programmable hypertext as a shell for CALL programs in *Computer Assisted Learning*, edited by L. Tomek, Springer Verlag.

Educating in the Hyperzone

Jack Wolcott
School of Drama DX-20
University of Washington, Seattle, WA, 98195, USA
wolcott@u.washington.edu

Abstract: This paper examines the disparity which often exists between what students expect of a college education and what they imagine that they have received. It proposes undergraduate involvement in the creation of hypertext and hypermedia documents as a means of synthesizing study in the core curriculum with study in fields of specialization, and provides examples from the development of "The Philadelphia Project," a hypermedia text.

The learning theory of William G. Perry, Jr. is introduced as the philosophical foundation upon which development and design of several hypertext and hypermedia projects in the School of Drama has been built.

Introduction

U.S. participants in the Viet Nam war reduced the earth's geography to two coordinate positions: "the World" – everything outside Viet Nam – and Nam, "the Zone." The distinction was clear cut and comforting: the World was real; the Zone was unreal. Higher education today presents a similar duality. Outside the institution is "the Real World," the place to which, after a short tour of duty in the "education zone," students will repair to spend the remainder of their lives, often embittered by the costly four years they have spent in-country, finding little to use from their experiences in the Zone.

It is easy to blame students for this situation, but the cause of this duality lies to some extent within the institution itself. The student approaches higher education with an expectation that he or she will discover interesting, relevant and, above all, useful information; useful because of its content, its coherence and its connectivity. Higher education clings tenaciously to an essentially eighteenth century model of the educated citizen, however, a model predicated upon the notion that a body of knowledge exists to which one should be exposed if one who would be called educated. This view is especially common in liberal arts programs, which insist the student become acquainted with a "core curriculum," usually made up of English literature and composition, history or philosophy, some mathematics and physical sciences, perhaps a little economics and a year of foreign language.

Information in the university data pool is overwhelming in its volume and complexity. There is no author in the hyperspace of higher education, no one who makes the links. Conspicuously absent from the core requirements of many liberal arts programs is Synthesis 101, a course to link the various disciplines studied, which perhaps could help the student find the links between changes in demographics and consumer spending, for example, or between the breakdown of monarchies in Europe and the rise of Romanticism in novels and dramatic literature.

The absence of an authoring agency in higher education is often reflected in the bewilderment of students who feel they have been forced to take courses which have no relevance to their life goals and interests. "All I ever wanted," said a young woman with whom I spoke recently, "was to be a theatre major. Instead, I wasted two years studying ocean sedimentation and calculus and other subjects I will never use again."

The Hyperzone

All of which brings me to the "hyperzone," where creating hypermedia becomes as important a learning tool as using hypermedia. Imagine a great plain, like something out of Rod Serling, littered with bundles of information. People wander aimlessly among the data bundles, picking out a bit here, a fragment there. In the center of the plain stands a portal. Pass through the portal and enter the world of Hypermedia, which promises to bring order to information bundles, to provide pathways between related bundles, to enable the pilgrim to explore new frontiers with confidence. "We cannot bring order to the other side of the portal" the inhabitants of Hypermedia tell the newcomers, "but we can show you how the bits and fragments you have acquired along the way relate to what we have here. You will have to help us, though, perhaps even retracing your steps to bring more things from the bundles you have opened on the other side of the portal."

In this model, hypermedia becomes a process, rather than a product. In the "hyperzone," activity on each side of the portal is of equal importance. Indeed, the hypermedia artifact cannot exist without input from the chaos of the liberal arts. The preparation of the hypermedia document becomes the synthesizing process so often found missing in higher education today.

At the University of Washington School of Drama's Center for Explanation, the name for which is taken from an article by Scott Kim in "The Art of Human-Computer Interface Design," (Brenda Laurel, editor, 1990) we have been conducting research for the past several years to integrate the widely scattered course work of the core curriculum into the study of theatre history. The discipline of theatre history, essentially a product of the twentieth century, has been characterized from its inception by its insularity. Until quite recently, the student of theatre history learned of actors and directors, of plays and playwrights, and of theatre buildings and theatre companies. Rarely was the theatre presented in a social, economic or political context, however. Indeed, one came away from the study of the theatre's history with the sense that this art existed in splendid isolation, free from the censorship of Oliver Cromwell, the rigors of the French Revolution and the persecutions of the McCarthy era.

Hypermedia and Theatre History

The Beginnings

In 1988, a colleague and I determined to approach the discipline from a new direction. Our plan was to involve undergraduates in the development of an open-ended hypermedia document, building the document out of student's research and scholarship.

The seed of this project had been planted over a year before, when a number of students who had taken a CADD class with me asked if there was anything they could do to hone these newly acquired skills. I proposed that they assist me in developing a tutorial on "Florimène," a seventeenth century English court entertainment, which would be used by students in beginning theatre history classes.

Working with the primary evidence from the English court in London – mostly drawings and written accounts of the extravaganza – the students found it necessary to explore seventeenth century drafting and construction conventions, and to research the politics of the court, the writing traditions of this entertainment form, and the music, clothing and court protocols as well. In short, they turned outside the discipline of theatre history to resolve problems within the discipline. Moreover, they became immersed in the processes of scholarship, finding and evaluating evidence, relating one piece of evidence to another, and determining how best to sequence and display information so that others might learn through the experience of using the "Florimène" courseware.

"Florimène" consists of fifteen short tutorials, which are accessible from each other and from an index as well. The user may move freely among the tutorials, or abandon them entirely and search out specific aspects of the entertainment through the index.

The successful development of the "Florimène" tutorial, which was produced using IBM's "Storyboard," suggested that rather than learning by using a computer application developed by someone else, students could also learn by creating an application themselves. Moreover, in the process of developing the application, students could be guided to make connections between what they had learned outside the department and what they were studying as theatre majors. Parenthetically, but of no small significance, the application developed in this process could then be used by students in other programs in the School.

William G. Perry, Jr.: A Philosophical Basis

The structure of the courseware and the enthusiastic reception of "Florimène" among the eighty students who served as the test group for the program was of special interest. Much of the computer assisted instructional (CAI) material available in 1988 was designed with a single learning style in mind, often consisting of drills, with branching and remediation. Students who learned by drawing inferences from primary source materials, or who learned by looking at numerous examples and formulating conclusions, had little material to which they could turn. Was it possible, we wondered, to create a single learning application which could serve numerous learning styles?

The research of William G. Perry, Jr. at Harvard University suggested an approach to CAI which seemed promising. Between 1954 and 1963 William Perry, who was director of the Bureau of Study Counsel, and his colleagues undertook research to discover how students learn in the college years. (Perry, "Forms of Intellectual and Ethical Development in the College Years," 1970) Perry suggested a developmental model of nine positions. At one end of a continuum is what Perry calls the "dualistic" learner, the student whose intellectual frame of reference is in terms of right and wrong answers. Challenged by ambiguity and supported by structure, the dualistic student can be moved to a new position, what Perry calls the position of early and late "multiplicity." Here the student begins to understand that not all knowledge is known, and that there are perhaps areas in human experience where answers are unknowable. Slowly, the student begins to rely upon his or her own powers of intellect, rather than upon the work of others.

Finally, students may move to what Perry calls the positions of "contextual relativism" a position in which the student must examine evidence, draw conclusions from the evidence and the circumstances in which it is found, and make moral and ethical decisions regarding how to act upon his or her conclusions.

The Philadelphia Project

Research and Development

With the experience of "Florimène" behind us, and Perry's work providing a theoretical base from which to proceed, our goals as we began work in the fall of 1988 on what came to be known as "The Philadelphia Project" were to develop a hypermedia application which would provide different learning environments for various learner types, and to use students in the development of this application, engaging them in problems that would encourage them to draw on what they had learned in the core curriculum for solutions. Owl International's "Guide" authoring software was the centerpiece for this activity. The development platform is currently an Intel 486 machine, running at 50mh, which controls a Pioneer LD-V6000 laser disk player with output to a 21 inch Panasonic video monitor.

The decision was made to focus the project on the city of Philadelphia and the Chestnut Street Theatre during the period between 1790 and 1835, an era in which I have been conducting research intermittently since 1964. Culturally, socially and politically, Philadelphia in the Federal period was fascinating. For a brief period of time it was the Federal capital, it had a rich cultural tradition, was the center of strong feelings both against and in favor of the theatre, and had a constant infusion of emigrants – escaped Blacks from the South as well as newly arrived Europeans. Moreover, the Chestnut Street Theatre was the United States' first theatre building expressly constructed for a professional company. Its company enjoyed a long and prosperous tenure in the city.

Hypermedia seemed an ideal environment in which to manage the fragmented data which survives the Chestnut Street Theatre, often only a sentence or two, a scrap of paper with a sketch, or an engraving or watercolor. Computer management of the primary evidence would permit weaving together into a coherent and accessible whole the bits and pieces which otherwise virtually defy organization and presentation.

We began with an undergraduate seminar of about fifteen students. This seminar had as its goal to explore questions of how people learn, and of how CAI materials might be developed which could accommodate various learning strategies. The first half of the seminar focused on Perry's schema. In the second half of the seminar each student developed a short tutorial which would be appropriate for a specific learner type. Special effort was taken to build in challenges which would prompt each learner type to move toward a new position. The seminar concluded with sessions in which attempts were made to modify and integrate tutorials in such a way that a single tutorial might serve several learning styles.

A second seminar saw students, many of whom had been in the earlier class, working together to develop a storyboard of ideas from which the final project would be developed. Each student was encouraged to suggest topics relating to the theatre, from which "brainstorming" sessions would begin to suggest broad links to the key topic. A student with an interest in theatrical costume design suggested "costume," for example. This led to a separation of "theatrical clothing" and "street clothing," which in turn led to questions such as where clothing came from in the early 1800's; whether the U.S. was producing its own textiles or whether these were being imported from England and the Continent; who did the styling and sewing of clothing, and how fabrics were dyed. Students who had studied economics raised questions regarding tariffs, which carried discussion to the various embargo acts which gave rise, in part, to the American Revolution. Finally, questions were raised concerning the extent to which clothing was related to social and economic class: how clothes were signifiers and what they signified.

In another example, "stage lighting" was proposed. This prompted questions of how homes, theatres and other public buildings were illuminated during the period of the study; how gas lighting was introduced to the theatre and how gas was generated; and of what the aesthetic effects of lighting with gas versus lighting with candles and oil lamps? Another line of inquiry developed around the question of how painting techniques had changed with the introduction of gas lighting, both because the viewer could see more clearly under the new illuminant and because many pigments blackened in the presence of the sulphur in the gas. The production of illuminating gas raised questions about the chemistry of distillation; about what role coke – a byproduct of the gasification process – played in the economics of manufacture in Philadelphia; and about the sources of bituminous coal and how it was transported to the city.

Once a basic storyboard of ideas had been proposed, students selected a topic and spent the second half of the seminar doing background research. While all these topics have their origins in theatre applications - costuming, candle and gas lighting, for example – the strategy of the seminar was to turn all inquiry outward. Not only did this encourage students to consider the institution of the theatre in the broader context of its parent society, but it encouraged students to draw upon their educational experiences outside the drama curriculum in an attempt to position the theatre relative to the society it served. To draw on my earlier metaphor, students were at work in the hyperzone without yet being at work with hypermedia tools.

Preparation

Owing largely to student enthusiasm for these topics, the Chestnut Street Theatre building and clothing were chosen as pilot projects for the construction of a hypermedia document. It was clearly understood by all involved that this was an open-ended project, one which would be concluded by entropy alone. Additional topics could be added as enthusiasms arose.

The actual accumulation of data for the hypermedia text took several directions. A three dimensional CADD model of the Chestnut Street Theatre building was developed. Students working on this project had only a measured elevation of the building's facade and a plan of the first floor. Additionally, the building, which burned to the ground in 1820, is represented by sketches on the back of an envelope and on bits of paper, and by the fragmented comments of actors and others who worked at the theatre.

Reconstructing an historical building by developing a three dimensional model such as this requires essentially the same skills and understanding of architecture, construction and engineering that would actually have been required to build the theatre in 1794. Solving the problems they encountered led students to an understanding of the aesthetics of Palladian and Georgian architecture, as well as to an understanding of eighteenth century construction practices; the economics of the tontine system, under which the building was financed; and the social and political climate of eighteenth century Philadelphia and how this affected physical arrangements in the theatre building.

Meanwhile, and this "meanwhile" lasted over eighteen months, undergraduate and graduate students were at work preparing materials for the clothing portion of the project. These preparations included the rehearsal and production of live action footage which would illustrate nineteenth century stage settings with actors in period clothing moving about, the development of tutorials which would accompany the clothing and costume pictorial data, and the production of a laser disk. Much of this effort was carried out in conjunction with staff from the Historic Clothing Collection of Seattle's Henry Art Gallery. Drama students thus had a unique opportunity to work with professionals whose expertise led them to understand clothing and costume in ways uniquely different from the understanding of the theatrical costume practitioner.

A graduate student in theatrical costuming prepared a series of costume drawings which depict the silhouette and overall characteristics of clothing in each decade from 1790 to 1850. Clothing which corresponded to these plates was selected from the Henry Art Gallery and from the School of Drama theatrical costume collections by students and staff. Tutorials based on this material were roughed out by a group of graduate and undergraduate students. The tutorials paid special attention to the style, construction and ornamental details of each garment, and a shot-list was prepared from this work for use later when the clothing was readied for photographing. Finally, prints and drawings which supported the clothing tutorials were identified.

During this research and preparation phase a doctoral candidate in the School of Art mounted an exhibit for the Henry Art Gallery which traced the development of the Kashmir shawl in the nineteenth century. This exhibit focused not only on the art of weaving illustrated by this medium, but also on the shawl as a socio-economic indicator in society. Upon hearing of "The Philadelphia Project," the curator of the exhibit not only offered the opportunity to photograph the exhibit for the laser disk, but agreed to write a tutorial as well.

The involvement with the shawl exhibit led to a surprise offer by a clothing historian whose specialization was shoes. He developed a pictorial database of period shoes for inclusion on the laser disk, as well as a tutorial based on this material.

In 1992, nearly a thousand images were recorded to video tape in preparation for producing the laser disk. These included over-views and close-ups of each garment, shawl and print so that, in effect, a viewer may examine every detail of the piece as though it were in hand. Additionally, full-motion sequences show what appear to be actors on the nineteenth century stage. In reality, these sequences consist of actors from the School, in period costume, chroma-keyed into the settings of a three foot by four foot toy theatre. Undergraduate students were involved throughout with the decision-making process which led to the video disk, with the video shoots and with editing the master tape. With this data now available to them, both the Chestnut Street Theatre building team and the students working on the clothing database were ready to pass through the portal into the computer managed portion of the hyperzone.

Construction of the Hypermedia Document

Since there is no closure, no urgency to create a finished product to be hurried off to market, design in "The Philadelphia Project" has been evolutionary rather than prefigured. The underlying principles which have informed choices in design have been those of William Perry.

The project has been shaped so that users in need of structure may explore the entire database through tutorials. Expert users may choose to move through the database making use of indexes and search-and-browse structures, ignoring the tutorials entirely. The user in the middle ground might choose to begin in a tutorial on seating in the theatre auditorium, move out to explore the index to the shawl collection and women's dress in 1830, become lost in this rich visual environment and turn back to the tutorial for help.

Student developers are strongly encouraged to experiment: there are no mistakes, only choices that don't work as well as others might. Nowhere is this to be seen more than in interface design. The "look and feel" of this project has evolved over time, driven by the materials which are constantly being added to the project, by student's changing understanding of relationships within the database, and often by intense debate over how to structure links.

The problem of integrating the 3D model of the theatre building into "The Philadelphia Project" illustrates the typical evolution of an interface. Ideally, the model would be presented so that it could be rotated in real time, with every room in the building buttoned. The viewer would merely stop the rotation, click, and examine the history, uses and artifacts of the selected space. Failing this, the model might be kept in 3D CADD, so that it could be rotated easily for viewing.

Software and budgetary considerations do not permit either of these solutions, so students developed a series of bitmap drawings from the model, created an index, and linked the index to the drawings. This solution had the weakness of leaving the various parts of the building dissociated from each other. Although links in the drawings associated each graphic with other parts of the database, this was essentially a flat solution to a dimensional problem.

Anticipating difficulties in transferring a 3D vector drawing to a 2D bitmapped world, the CADD model was designed so that each wall and floor is on a separate layer. It is therefore possible to display the model from any rotation, removing walls and floors which are in the way. Thus, it was possible to create a kind of cell animation, or mosaic of views so that, in effect, a person can do a "walk through" or "fly around" of any portion of the building. After experimentation with "Guide," a solution was developed which places numerous drawings together on the screen. Each drawing is in a separate borderless window and all windows on the screen are simultaneously active. Careful placement of windows on the screen permits comparisons of various views. Buttons abound in each picture. Some buttons link to close-up details of the drawing, while others bring prints and drawings to the video monitor, or provide the user access to tutorials and indexes in which the user may find associated data.

The most important lesson to emerge from this experiment has been to understand the necessity for leaving loosely defined the shape of the hypermedia artifact. In architecture, form follows function. In the hyperzone, form follows problem solving and the discoveries which result from students synthesizing information into knowledge and knowledge into understanding.

"The Philadelphia Project" continues, changing shape each academic term as new students become involved. It is probably safe to conclude that few students who work on the project realize they are working in what I have referred to as the hyperzone. Students approach the idea of creating a hypermedia text warily if the idea is presented head on. It seems too daunting a task. But confront them with a problem, challenge them to organize complex data in such a way that it is responsive to this problem, and the hypermedia environment becomes a safe haven, a place where organization and synthesis seem to take place effortlessly, a place from which sorties may be made out of the comfort and safety of the major discipline and back into the intellectually challenging chaos of the liberal education.

Using a Computerized Relational Glossary for Syllabus Design and Student-Centred Study

MARK WOODMAN and GORDON DAVIES, *Computing Department,*
BOB ZIMMER, *Institute of Educational Technology,*
The Open University, Walton Hall, Milton Keynes MK7 6AA
United Kingdom
Email ids: m.woodman@open.ac.uk, g.davies@open.ac.uk, r.s.zimmer@open.ac.uk

Abstract: This paper describes a hypertext application to assist with the design and study of courses by using a software implementation of a relational glossary. It is based on the established view that topics are made up of related concepts and that their relationships can be captured in a relational glossary. An entry in a relational glossary contains one concept as a grammatical subject, together with explanations for this concept consisting of grammatical predicates involving other concepts. These set the concept into the context created by other related concepts. The tool is a HyperCard stack with a well designed interface to enable the production and interrogation of the stack. The relational glossary can be used by course designers to check the relationships between the concepts covered in a course and by students to check that they have understood particular concepts and the relationships between them. In addition, when students become stuck, they can use the glossary to get themselves moving again.

Background

The Open University (OU) is the United Kingdom's university for distance learning. It also offers its programmes of study throughout continental Europe. In 1992 there were 136 courses in the OU's undergraduate programme and in this academic year over 126000 students have registered to study with the university. The degree programme is modular and there are relatively few constraints on how students can mix courses and subject areas.

The distance mode in which the Open University operates has a significant impact on its syllabi and its pedagogy. OU students are taught using correspondence texts, broadcast television and radio, and video and audio cassette programmes. The first of these is the main pedagogic vehicle. The relative isolation of students from their peers and tutors is important; there are no 'lab' sessions to help students, and so careful syllabus design is vital. It is also imperative that students develop independent learning skills and be provided with the means to use these skills. Therefore, in this section we set the context which has motivated us to design a computerized version of an artefact which assists syllabus design and student-centred learning.

The average age of students is 34 (mainly in the 25–45 age group) and the vast majority remain in employment throughout their studies. This makes courses that can be considered vocational very attractive to OU students; however, study is part-time and demanding. Computing courses are particularly popular for their vocational potential; the Computing Department's courses are much in demand, with over 2500 students per year taking the introductory undergraduate course alone, despite it requiring at least 400 hours of study over ten months. We have just begun the development of a new introductory course and this is the primary motivation for the work reported herein.

The current introductory course covers many topics in programming, software engineering, software systems, structured analysis and design, and the social impact of computing. Because of its level, certain assumptions are made about the ability (the self-sufficiency) of students. To some extent this has meant that the authors of the course were able to avoid some teaching of elementary study skills; however, changing regulations and governmental demands means that, increasingly, too many students do not have such skills. Furthermore, as is usual with OU courses, the introductory course for computing has no prerequisite and attracts students who have substantially different requirements of the course. For many it is the first of a number of computing courses which will build a computing profile for their degree. For others the course is a diversion from arts, the pure sciences or social sciences; in other words it may be used as an appreciation course. The consequence of this mix and the variable level of study skills is that any replacement must be accessible to a wide diversity of students following different academic careers. These interacting factors mean that the new course will have to provide greater assistance to students to enable them actively to explore the course syllabus, whether it be to take remedial action or to deepen their knowledge.

Relational Glossaries

One device which can be used to enable students to explore the syllabus to deepen their knowledge or take remedial action is a relational glossary – a glossary which defines the concepts in a topic in such a way as to make the relationships among all the concepts very clear (Zimmer, 1984). Before discussing how we might create such a glossary, we give three operational definitions as specified by Zimmer:

- A topic is defined as 'an agglomeration of concepts that are meant to be inherently interconnected.'
- A concept is 'anything that can be named as a noun.'
- An explanation is 'one or more sentences in which the name of the concept being explained appears as the grammatical subject.'

In addition, the relational structure of a topic must be represented in a way that is 'capable of displaying the three properties of uniqueness, completeness and connectivity' (*cf.* Pask, 1975):

- *Uniqueness*: Once one has named a concept in the topic, one never gives it another name and one never uses its name for any other concept in the topic.
- *Completeness*: One can explain any named concept in the topic in terms of other named concepts in the topic without recourse to concepts outside the topic (except for those in the everyday language of the learner).
- *Connectivity*: A thorough explanation of any concept in the topic presents several aspects of its meaning in terms of several other concepts in the topic.

TERM	RELATIONSHIPS	UNIT	SECTION
SYNCHRONISED CLOCKS at rest in any particular **inertial frame of reference**	can be thought of as all running at the same rate and reading the same **instant in time** no matter where they are. That is, they would all appear to read exactly the same instant in time for a hypothetical, *non-physical* being ... They are therefore exactly the same in concept *both* in Newtonian mechanics and in **Einstein's special theory of relativity**, *provided that* one thinks about them only within a *single* chosen inertial frame of reference.		
	in **Einstein's special theory of relativity**, will *not* appear to be synchronized ... (See the **relativity of simultaneity**.)	5	4.2
	in Newtonian mechanics, are *assumed* to appear synchronized *regardless* of the frame of reference from which they are observed.		
	in **Einstein's special theory of relativity**, are embodied in a **space-time diagram** by the very *existence* of an x-axis or an x'-axis. ...	5 7	7 4
	can be set up, in **Einstein's special theory of relativity**, by a simple procedure which allows for the finite **speed of light** . .	5	4.3
A **TEST CHARGE**	is a particle bearing a known *small* **charge** (usually called charge *q*) which is used to measure the **electric field** at a point in **space-time**. (The charge on it must be small so that its effects on the charge distributions or current distributions that give rise to the fields being measured can be considered negligible.)	4	2.2
The **THRESHOLD ENERGY** for a reaction	is the minimum **energy** required to start the reaction. This energy is usually supplied by the **relativistic energy** of an incoming particle.	7	6.5

Figure 1: An extract from a paper-based relational glossary

In Figure 1 we give, as an example, an extract taken from an Open University course in physics (S354, 1980). In this example, each term which appears in bold print can itself be looked up in the glossary.

Such a glossary can be used by students for exploration, revision and review. All that they need to do is to browse it and follow its rich cross-referencing.

More importantly, they also can use it to get 'unstuck' when – inevitably – they get stuck from time to time while learning from the main course text. All that they need to do is to write down the question to which they need an answer in order to get moving again – and then look up in the glossary the concepts which appear in their question. Provided that the glossary has been constructed with proper uniqueness, completeness and connectivity, the explanation needed will appear immediately or else through tracing of cross-referenced terms.

Students are not the only ones to benefit from a relational glossary. While they benefit from its completed form, the very act of constructing it can be of huge benefit to teachers when designing a course – particularly where the subject matter is complex (Zimmer, 1984) and awash with jargon, as in computing. For the educator, a relational glossary encapsulates the set of topics to be covered by a course although it does not order them according to some pedagogy. Indeed, any ordering is determined by the contents of the glossary. While this benefit has been recognized, there has been a reluctance to invest in the time and effort needed to create comprehensive relational glossaries; pressure to 'get writing' often precludes their use during syllabus design.

We conjectured that a computer-based relational glossary would encourage academics to use it as an experimental tool and that it could become an effective computer-based learning tool for students. Desire for such a tool has been expressed in other parts of the OU as well (Clark, 1989). We are in the process of testing these views.

Student Use of a Computerized Relational Glossary

A paper relational glossary can be of use to both course developers and to students, but a computerized relational glossary potentially even more useful. Firstly, a teacher can use it to record the specification of a course syllabus and to check it automatically for coherence. Secondly, with appropriate support, a teacher can experiment with and design syllabi based on recorded concepts. A prototype system has therefore been developed based on HyperCard; for the introductory software development course, a production-quality system will be written using the prototype as an oracle. In this section we describe the prototype as it might be used by a student. In the next section the computer-based system is described in terms of creating the glossary. For brevity we do not describe how a user is determined to be a student or a teacher.

The structure of the information held is a straightforward organization consisting of an unordered sequence of triples containing the *concept name*, a concise *explanation* of the concept, and an unordered list of *related concepts*. Hyperlinks may be established between concepts referenced in an explanation and the descriptions of those concepts, and between related concepts and their descriptions.

It should be noted that this structure is 'looser' than the structure shown in Figure 1: for the purpose of creating our prototype, we have gone for ease of construction in preference to clarity of access. Once the system is working well, the more exacting format shown in Figure 1 can be introduced.

The HyperCard prototype consists of a 'stack' of cards containing corresponding fields as shown in Figure 2, which shows a card from a computerized glossary called 'M206 Glossary'. The concept (or term) to be described is shown in a banner heading style; in Figure 2 it is *Computer System*. The explanation is given in the below (beginning *This is the sum total...*). A list of related concepts may be shown in a narrow field below the explanation (see Figure 3); these are terms not needed in a concise explanation of *computer system*, but should be explored for a full understanding. These related topics are only shown to to a student on request, by clicking the *Show Related Concepts* button shown in the figure.

The explanation of *computer system* contains references to three other concepts; these are *operating system*, *software system* and *system*. Clicking on any of these will transfer the user to the appropriate card containing the description of the reference term. For example, clicking on *software system* moves to the card shown in Figure 3.

Normally, the text in the glossary fields cannot be modified; however, a panel of buttons for editing them and for reading from text files or other glossaries can be used using a *Show Control Panel* button. This button is not shown in Figure 2 as students would not usually see this button; it is only available to users who are teachers. In addition to the glossary fields, a typical card contains a field (on the right) where a trace of browsed terms is maintained. The empty trace in Figure 2 indicates that no concept was browsed before *Computer System*. Figure 3 shows *Computer System* in this field, indicating that its description has been the only one visited during the current browse. A card also has buttons for navigating whose functions are indicated in Figure 2.

In the prototype, the introductory card of a glossary, though of the same general structure, is used differently from the rest. It is used as a general *information* card and contains additional fields and extra operations, provided by buttons (these are mostly of use for the author of the glossary as described in the next section). An example from the glossary being used to develop a new syllabus is shown in Figure 4. There, general information is given in the largest field, as shown.

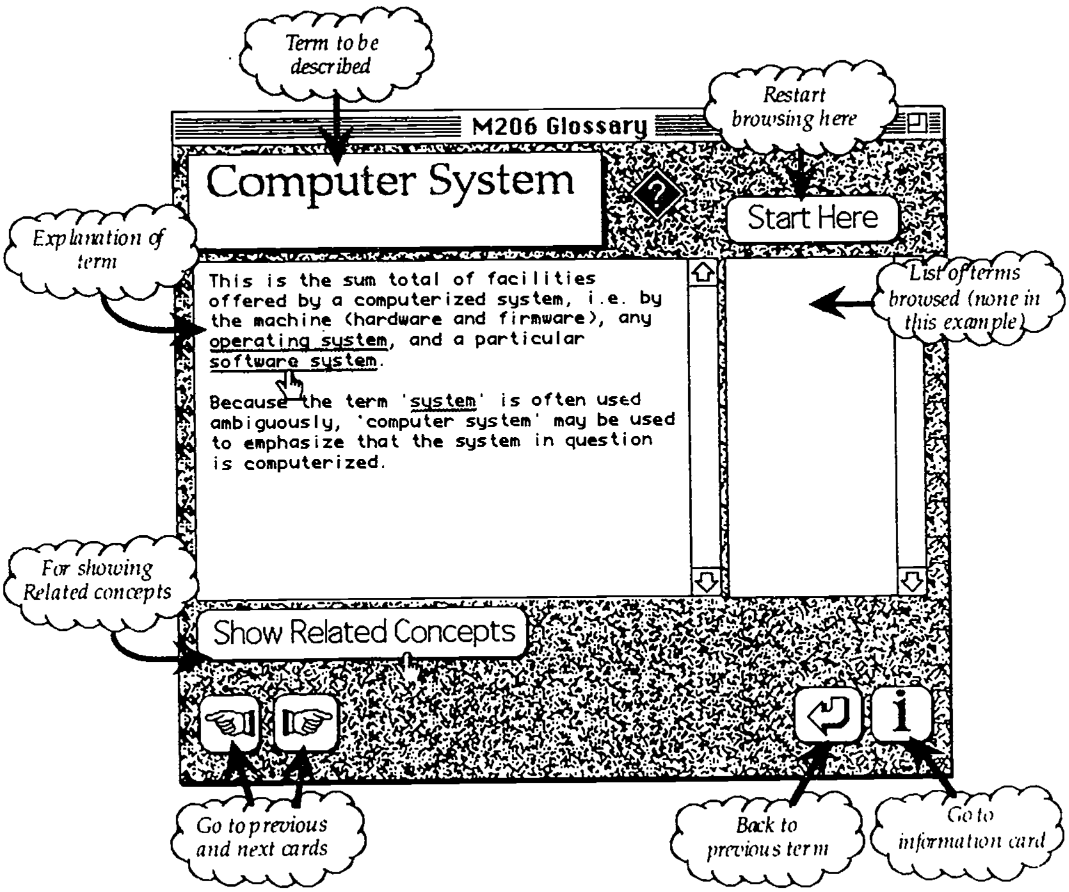


Figure 2: Typical card from computerized relational glossary

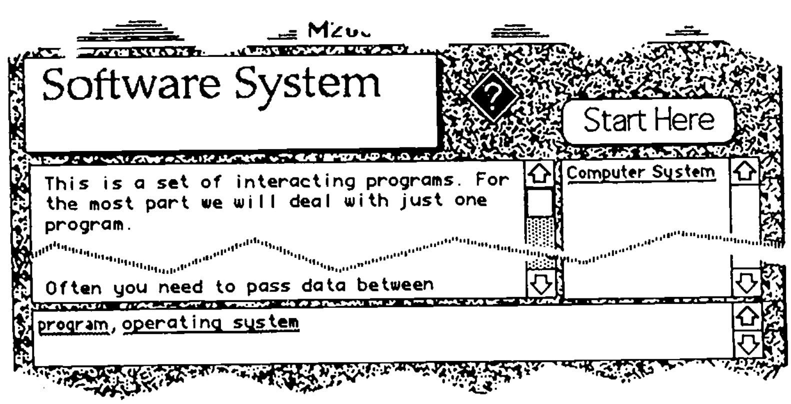


Figure 3: Navigating a network of concepts

Note that the right hand field in the information card is an alphabetically ordered index to all the concepts held in the glossary. A term can be found by scrolling through the list (or using a built-in 'find' command) and the user can move directly to its explanation by clicking on the term.

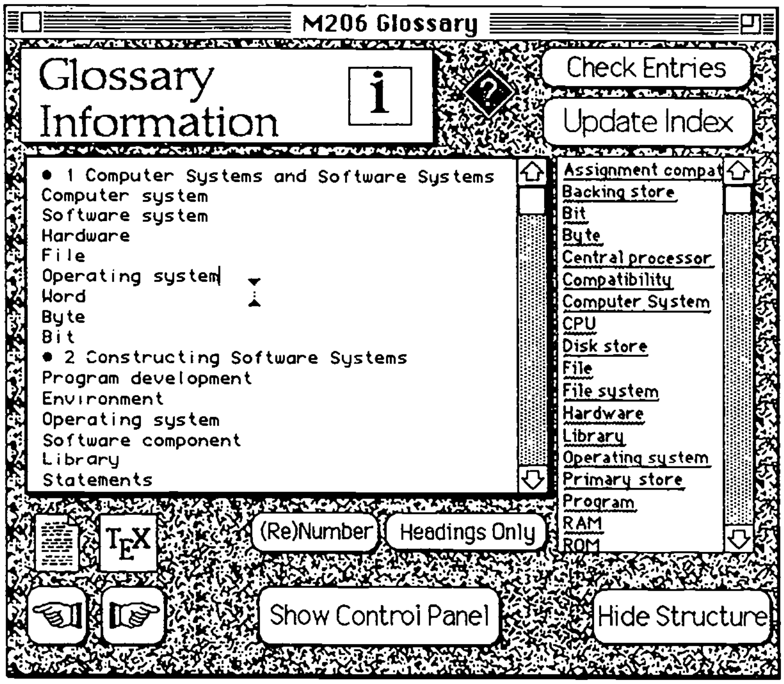


Figure 4: Sample syllabus structure

The buttons immediately under the large explanation field (shown by the general document icon and TeX) are for producing reports of the glossary according to parameters which may be set via the control panel mentioned earlier. The first generates a plain text description of all concepts, their explanations and related concepts; the text can be copied for pasting to another application or written to a text file. The second generates a LaTeX document (Lamport, 1986). The purpose of this is not just to produce an elegantly typeset document but to produce comprehensive cross references from explanations and related concepts and to produce various types of index.

Syllabus Design Using a Computerized Relational Glossary

The prototype has several facilities for experimenting with and designing a syllabus. Although a relational glossary is inherently unordered with a topic structure being like a well-woven string-bag which can be picked up and hung from almost any node, it is helpful to be able to order the concepts in some way. This facility and others are available from the *Glossary Information* card. The card now has a *Try Structure* button which, if clicked, reveals a field with a shadow style for recording concepts to be included and for ordering concepts. Also, the button changes to become the *Hide Structure* button at the bottom right of Figure 4. Using this field the syllabus designer can explore a two-level structure consisting of headings (prefixed by a bullet character) and concept names. Figure 4 shows an example of this structure in the larger field on the left. It may be edited in the usual way, so text may be selected and copied/cut and pasted. Concept names may also be copied into the structure by clicking on the index field on the right. In addition, the headings may be numbered (as in Figure 5) and the numbers recalculated or removed as desired using the *(Re)Number* button.

The report generation buttons operate somewhat differently when a structure is being developed: the user is given the option to report only on the headings, or headings and concepts.

When designing a syllabus it is often necessary to concentrate on the headings under which concepts are grouped to make a topic. The *Headings Only* button is therefore provided for use with the structure field; as the name suggests, it removes the detail of the concepts, leaving only a visual indication that concepts have been included under a heading. Figure 5 shows an unnumbered, elided version of the structure in Figure 4. The *Headings Only* button is replaced by an *Show All* button for restoring full details.

Another facility for the author of a glossary is an operation to check what concepts have not yet been entered in a glossary. The *Check Entries* button (top right, Figure 4) produces a report on the screen on topics not defined. Entries for these missing concepts may then be created automatically either en bloc or singly by clicking on the name of the missing concept. (This facility is also available for creating concepts enumerated in a structure such as that in Figure 4.)

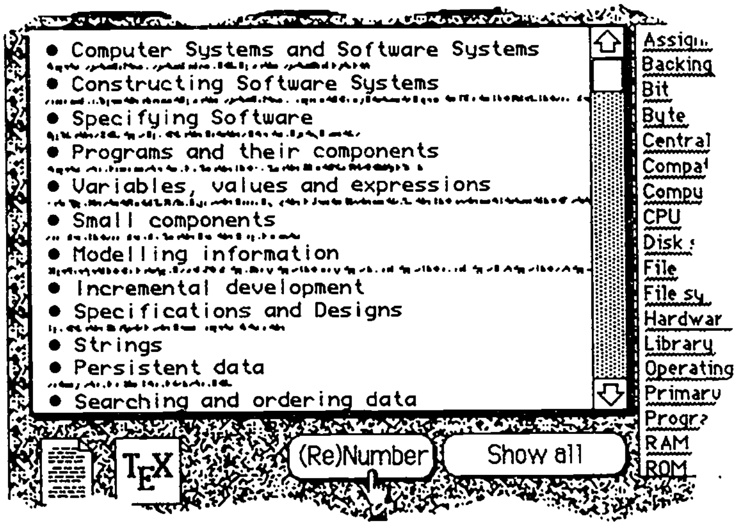


Figure 5: Elided, unnumbered structure of a syllabus

While using our prototype in the ongoing design of the new course we, as educators, have been confronted with the uncomfortable fact that we tend to introduce too many concepts early in our courses and that we often do not make good use of the concepts we have previously described. This experience supports anecdotal evidence that students find the early part of the existing course too onerous. It also shows that construction of a relational glossary during syllabus design may expose the tendency of experts to assume too much of the novice.

Conclusion

We have described a need for a computerized relational glossary for syllabus design and student-centred study and have outlined a novel tool to implement this notion.

A relational glossary can help students to revise and review and get unstuck while learning. The act of constructing it can help teachers to produce course materials of much greater conceptual clarity, whether working alone or in teams. The problem is that proper construction of a relational glossary takes time and effort. Computer software to prompt and facilitate its construction can make the process much more attractive.

The software is still in a prototype stage, but it is being used to design a syllabus for the OU's new introductory computing course. The glossary facilities have already been tested by reproducing part of an existing paper glossary of a programming language standard (ISO, 1992). The syllabus design facilities were tested by analysing and inputting to a glossary material from an existing computing course. As part of normal quality assurance process the production-quality version of the computerized glossary will be tested and evaluated by students before it is used by their peers.

We have also found that using a hypertext system for creating teaching material has indeed many of the benefits expounded by others (Thimbleby, 1992).

References

Clark, P. (1989). Private communication from Dean of Science, The Open University, to R. S. Zimmer.

ISO (1992). ISO/IEC JTC1/SC22/WG13, *CD 10514 Programming Language Modula-2, 2nd Committee Draft*, ISO/IEC JTC1/SC22 Secretariat, Ottawa, December 1992.

Lamport, L. (1986). *LaTeX: A Document Preparation System*. Addison-Wesley, Reading, Massachusetts.

OU (1980). *S354: Understanding Space and Time*. The Open University, Milton Keynes, UK.

Pask, G. (1975). *Conversation, Cognition and Learning: A Cybernetic Theory and Methodology*. Elsevier.

Thimbleby, H. (1992). *Thinking Creatively with Hypertext*. University of Stirling Computer Science Technical Report TR 95, October 1992.

Zimmer, R. (1984). The relational glossary – an aid to learning and review. In *Independent Learning in Higher Education*. E. S. Henderson, M. B. Nathenson, eds., Englewood Cliffs, New Jersey. pp 125–131.

Curriculum Knowledge Representation in SQL-TUTOR

GANG ZHOU
Department of Computer and Information Science
New Jersey Institute of Technology
Newark, NJ 07102

Abstract: The research in the area of Knowledge-Based Tutoring System (KBTS), or Intelligent Tutoring System (ITS), has great importance for computer applications in education and training. The objective of this research is to develop a framework upon which effective KBTSs can be built. We have developed SQL-TUTOR, a KBTS for the domain of SQL programming, which adopts a new architecture called *curriculum oriented* in the sense that both domain knowledge base and student model are built based on the course curriculum. The course curriculum is represented by a *curriculum graph*, which incorporates the knowledge about the goal structure of a course, multiple viewpoints of a goal, and prerequisite relations among different goals. This new architecture has such features as i) explicit curriculum knowledge representation; ii) multiple teaching sequences; iii) multiple viewpoints of a teaching goal; and iv) overall skill training to a student.

Introduction

The goal of KBTS research is to build instructional programs that incorporate well-prepared course material in lessons that are optimized for each student. In a KBTS, course material is represented independently of teaching procedures, so that problems and remedial comments can be generated differently for each student. Such a system can offer instructions in a manner that is sensitive to the student's strength, weakness, preferred style of learning and provide individualized tutoring. In the design of SQL-TUTOR, we have adopted an architecture based on layered curriculum to organize system components[Chan, 1992, Lesgold, 1987]. The idea is to capture a teacher's knowledge of the curriculum at the first place and make this knowledge explicit to the system. We believe that this architecture strongly supports the perspective that views a KBTS as a knowledge communication system[Wenger 1987]. Figure 1 shows the components in our system.

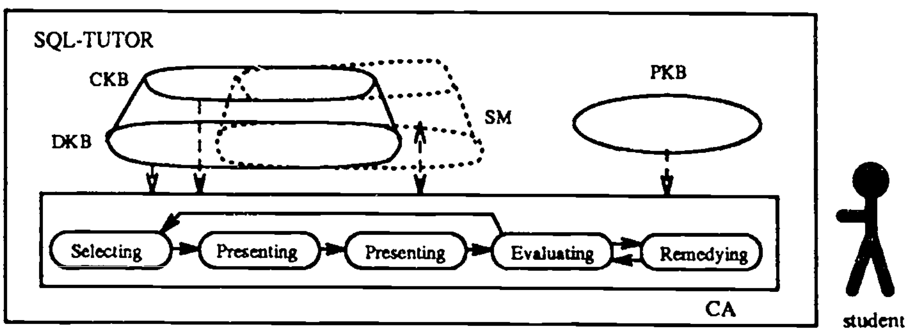


Figure 1: SQL-TUTOR

In this architecture, the knowledge in a KBTS is decomposed into four components and stored in Curriculum Knowledge Base (CKB), Domain Knowledge Base (DKB), Pedagogical Knowledge Base (PKB), and Student Model (SM). Another component in SQL-TUTOR is the Commuı ication Agent (CA), which controls the interactions between SQL-TUTOR and a student. It consists of five modules: i) *selecting module* - selects a new topic for the student to study; ii) *presenting module* - choses the piece of the text to be presented and display it to the student; iii) *answering module* - reads and answers the student's questions; vi) *evaluating module* – evaluates the student's performance; and v) *remedying module* – finds a remedical decision and applying it. In the following sections, we will discuss the design of the CKB in detail.

Content Tree and Simple Curriculum Graph

Suppose you are working on editing a textbook. How would you like to organize the subject materials that you are going to teach your students? Most people will organize them by chapters, sections and subsections, and use a contents table to list their titles. This is, in fact, a *decomposing* process. A book is decomposed into chapters, a chapter into sections, and a section into subsections. From a content table, a student can know the structure of the book: chapters, sections, subsections, and their relations. Thus a content table conveys some kind of meta-knowledge, the knowledge about the subject matters. We call this type of knowledge ***curriculum knowledge*** .

Because it determines the selection and sequencing of topics in a course, a KBTS must formulate a representation of curriculum knowledge. As we switch from the educational perspective that we have just used in our discussion to KBTS perspective, two changes are noteworthy: i) The **subject materials** are the target knowledge that a KBTS tries to impart to a student and is represented by the **domain knowledge** in the system. ii) **decomposing** subject materials is exactly a **subgoaling** process on the domain knowledge, and the resulting ***goal-subgoal tree*** corresponds to the content table. We call this tree a *content tree* . The goal associated with the root of a content tree is the topic of the book (e.g. teaching SQL). At the next level are a list subgoals, which are the topics of each chapters (e.g. introducing SQL). The goals of each chapter in turn are split into subgoals, which are the topics of each sections (e.g. teaching SQL concepts), and so on. The subgoaling process terminates at some lower level when the domain knowledge has been split into such simple sections that they could be taught to the student completely as one unit. So a leaf of such a tree is also called a *unit*. To summarize, we have the following two definitions.

Definition 1 The knowledge about the domain knowledge in a KBTS is called *curriculum knowledge*.[]

Definition 2 A *content tree* for a course is a tree with the following features: i) each node is associated with a teaching goal (topic); ii) the goal associated with the root is the goal of the course; iii) the goals associated the children of a node is its subgoals (or in another word, the goal at a node consists of a list of subgoals.[]

Figure 2 shows a content table and its corresponding content tree. A node in a content tree is called a *goal node*, or simply a *g-node*. An edge in a content tree is called a *goal edge*, or simply a *g-edge*. A content tree is denoted by $CT=(G, GE, R)$, where G is the set of g-nodes, GE is the set of g-edges, and R is the root of the tree.

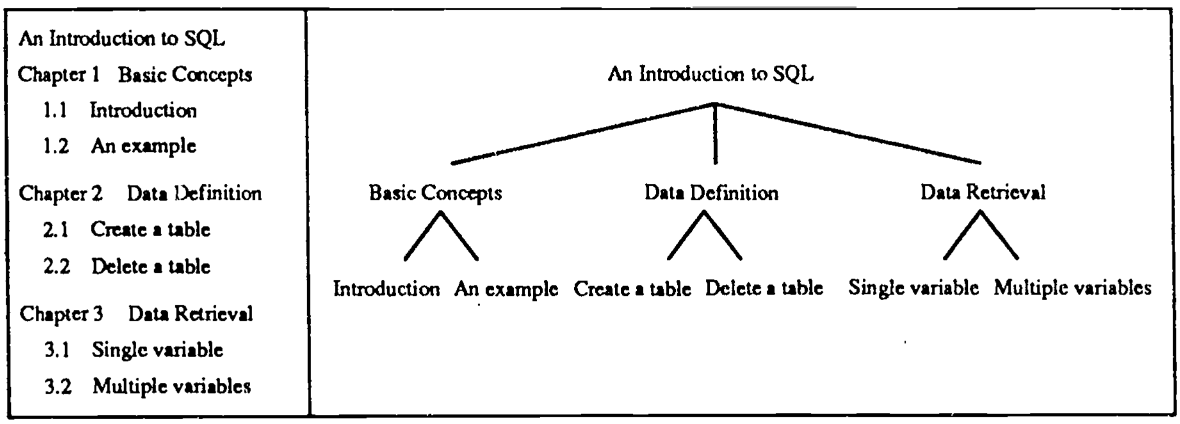


Figure 2: A content table and its corresponding content tree

Although simple, the content tree generated by the subgoaling is just a rough representation of curriculum knowledge in a teacher's mind. It suffers from two major weaknesses:

- *single sequence restriction:* the structure of the course is fixed, and there is only one sequence to go over the whole course. This restriction has the pedagogical meaning that a student can only follow one sequence to study the domain subjects and this sequence contains all the topics of the course. That is, a student has to study all the chapters and all the sections in a chapter in a predefined order to finish the course. Since generally there are several suggested ways to go through those topics of a course (some of them are required for all students, some of them are only appropriate for advanced studies, and some of them are only for readers with special interests), it is more desirable to organize them in such a way that the course could be easily tailored according to individual requirements of each student.

- *single viewpoint restriction:* each teaching goal is represented by only one perspective. This restriction allows the student to study the subject only in one perspective. Because things are not always so simple, a teacher might find that, depending on the different pedagogical purposes, there are different viewpoints on the instruction he wants to present. For example, there are three different viewpoints on the data objects managed by a DBMS: i) from the theory of relational database; ii) from relational algebra; ii) from file management system. Thus we have three sets of terminologies, {*relation, tuple, attribute*}, {*table, row, column*}, {*file, record, field*}, which refer to the same concepts.

In order to the lift first restriction, the knowledge about the *prerequisite* relations between the pieces of the domain knowledge can to added to a content tree.

Definition 3 Let $CT=(G, GE, R)$ be a content tree and $l_1, l_2 \in G$ be two leaves. Leaf l_2 is said to be *dependent on* leaf l_1, denoted by $\mathbf{dp}(l_1, l_2)$, if the goal at l_1 is a part of the background for teaching the goal at l_2.[]

Definition 4 Let $CT=(G, GE, R)$ be a content tree and two distinguished nodes n_1, n_2 be siblings. Node n_1 is a *prerequisite* of node n_2, denoted by $\mathbf{pr}(n_1, n_2)$, if either $\mathbf{dp}(n_1, n_2)$, or there exist two leaves l_1 and l_2 such that: i) $\mathbf{pr}(l_1, l_2)$; ii) l_1 is a successor of n_1, and iii) l_2 is a successor of n_2.[]

The prerequisite relation is the most important one among the topics of a course. It can be used to generate or check valid teaching sequences for a course in which a topic can be presented only if all of its prerequisites have been presented. It can also provide useful information to help the system to identify a student's missing conceptions or misconceptions in a diagnosing process. The prerequisite relation has the following properties: i) ***irreflexivity***: for all nodes n in a CT, $\mathbf{pr}(n, n)$ is false; ii) ***transitivity***: for all nodes n_1, n_2 and n_3 in a CT, if $\mathbf{pr}(n_1, n_2)$ and $\mathbf{pr}(n_2, n_3)$, then $\mathbf{pr}(n_1, n_3)$; In other words, it is a partial relation over the topics of a course.

Definition 5 Let $\mathcal{CG}=(G, GE, R)$ be a content tree and $n_1, n_2 \in G$ be two nodes. $\mathbf{pr}(n_1, n_2)$ is *primitive* if there not exists a node $n_x \in G$ such that both $\mathbf{pr}(n_1, n_x)$ and $\mathbf{pr}(n_x, n_2)$ are true.[]

If we add prerequisite information to a CT, the result is a lattice called *simple curriculum graph*, or $\mathcal{SCG}$. In a simple curriculum graph, there is a directed edge from node n_2 to node n_1 if and only if $\mathbf{pr}(n_1, n_2)$ and it is primitive. Such an edge is celled a *prerequisite edge*, or simply a *p-edge*.

Definition 6 A simple curriculum graph is a four-tuple $\mathcal{SCG}=(G, GE, P, R)$, where G is the set of goal nodes; GE is the set of goal edges; P is the set of prerequisite edges; R is the start node.[]

When selecting a new topic for the student, the system can select any topic as long as the following constraint is satisfied: ***if*** $\mathbf{pr}(n_1, n_2)$, ***then the presenting of*** n_2 ***must follow the presenting of*** n_1. Since this procedure is nondeterministic, the system can generate multiple teaching sequences.

Curriculum Graph

Sometimes, there are several valid ways to decomposed a goal into a list of subgoals. The different decompositions view the same goal from different perspectives. So, from any specific point of view, there are clear pathways that determine the sequencing of instruction, though of course there are other alternative sequences to approach the same goal. There may be psychological emotions which result in individual differences in aptitude or preference for different viewpoints.

As an example of multiple effective perspectives to view a teaching goal, consider the decomposition of `SELECT` clause of `SELECT` statement of SQL language. We can partition this goal according to the types of the data objects they applied to, resulting in two subgoals SINGLE-COLUMN-RETRIEVAL, MULTIPLE-COLUMN-RETRIEVAL; or partition them into those that have or have no qualifiers, resulting in UNQUALIFIED-RETRIEVAL, QUALIFIED-RETRIEVAL. If we incorporate this feature of multiple viewpoints to a simple curriculum graph, the result is a *curriculum graph*, or simply $\mathcal{CG}$. In such a graph, the children of a node may represent the multiple viewpoints about the teaching goal of the node. We call this type of node a ***multiple viewpoints node***, or simply a *m-node*, and an edge connecting a *m-node* and its children a ***multiple viewpoints edge***, or simply a *m-edge*.

Definition 7 A *Curriculum Graph* is a six-tuple $\mathcal{CG}=(G, M, GE, ME, P, R)$, where G is the set of goal nodes; M is the set of multiple viewpoints nodes; GE is the set of goal edges; ME is the set of multiple viewpoint edges; P is the set of prerequisite edges; R is the start node.[]

Figure 3 is a curriculum graph evoluted from the content tree shown in Figure 2, where Data Retrieval a m-node with two viewpoints: based on data objects (DBR) and based on qualification (QBR). Both of them are g-nodes.

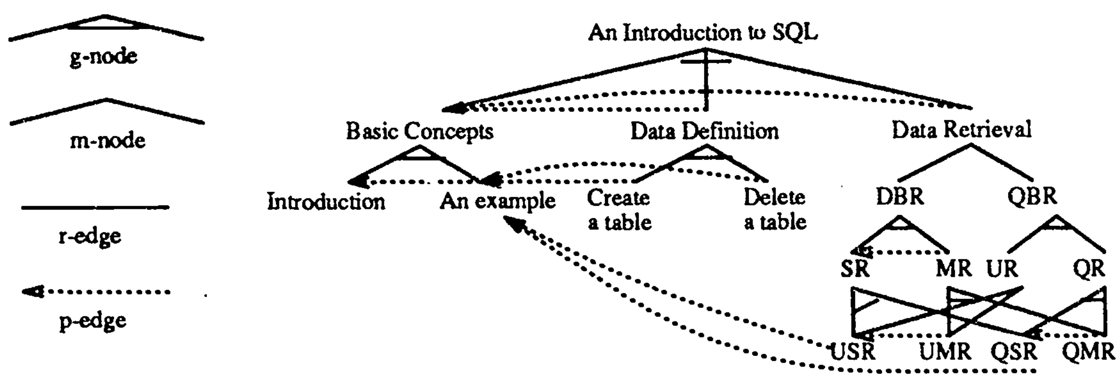


Figure 3: A Curriculum Graph

A r-node or m-node is also called a *regular nodes*, or simply *r-nodes*. A g-edge or m-edge is called a *regular edge*, or simply a *r-edge*. A node m is a *regular ancestor (successor)* of another node n in a $\mathcal{CG}$ if they are connected by a series of regular edges. In the curriculum graph of Figure 3, the regular ancestor of SCUS include these node: An Introduction of SQL, Data Retrieval, RBOD, RBOQ, SCR, UR. What is noteworthy is that the decompositions of the multiple viewpoints of a m-node should always produce the same set of nodes at some lower level. We call this set of nodes the *base* of the goal. As shown in Figure 3, the base of Data Retrieval consists of four nodes SCUR (Single Column Unqualified retrieval), MCUR (Multiple Column Unqualified retrieval), SCQR (Single Column Qualified Retrieval), and MCQR (Multiple Column Qualified Retrieval). Nodes in a base are not necessarily units. Some or all of them can be interior nodes. The following is the formal definition of a base.

Definition 8 Let $\mathcal{CG}=(G, M, GE, ME, P, R)$ be a curriculum graph, $m \in M$ be a m-node and $n_1, \ldots, n_k \in (G \cup M)$ be its children. The *base* of m, denoted by *base*(m), is the set of nodes with all of $n_1, \ldots, n_k$ as their only regular ancestors.[]

The requirement that all the children of a m-node map into the same base at some lower level in the curriculum graph implies two things: i) the child of a m-node can not be a leaf; ii) all the nodes in *base*(m) have k direct regular parents, if m has k children.

Study Graph and its Properties

Definition 9 Let $\mathcal{CG}=(G, M, GE, ME, P, R)$ be a curriculum graph, and $N_s = \{n_1, n_2, \ldots, n_k\}$ be a set of leaves. A *study graph* of N_s is a connected subgraph of $\mathcal{CG}$ with start node R and tipnodes $n_1, n_2, \ldots, n_k$. It is *well defined* if for any nodes n_i, n_j, if $\text{pr}(n_i, n_j)$ and $n_j \in N_s$, then $n_i \in N_s$.[]

Generally speaking, for a set of given nodes N_s, there are several study graphs, and we denote them by $\text{sg}(\mathcal{CG}, N_s)$. Obviously, the curriculum graph itself is the maximum well defined study graph of its leaves.

Definition 10 Let $\mathcal{CG}=(G, M, GE, ME, P, R)$ be a curriculum graph and $l \in G$ be a leaf. The *a-closure* of l in $\mathcal{CG}$, denoted by $\mathbf{a}^*(\mathcal{CG}, l)$, is the set of prerequisite nodes of l, or $\mathbf{a}^*(\mathcal{CG},l) = \{m|\mathrm{pr}(m, l), m \in G\}$. The *s-closure* of l in $\mathcal{CG}$, denoted by $\mathbf{s}^*(\mathcal{CG},l)$, is the set of nodes which take n as their prerequisite, or $\mathbf{s}^*(\mathcal{CG},l) = \{m|\mathbf{pr}(l, m), m \in G\}$[]

Definition 11 Let $\mathcal{CG}=(G, M, GE, ME, P, R)$ be a curriculum graph and. $n \in G$ be a node. The *learning graph* of n with respect to $\mathcal{CG}$, denoted by $\mathbf{lg}(\mathcal{CG},n)$, is defined as: i) if $n \in G$ is a leaf, then its *learning graph* is the *maximum* study graph of its a-closure, or $\mathbf{lg}(\mathcal{CG}, n) = \mathbf{sg}(\mathcal{CG}, \mathbf{a}^*(\mathcal{CG},n))$; ii) if $n \in (G \cup M)$ is not a leaf, then its *learning graph* is the *maximum* study graph of the union of the a-closures of those leaves which are the regular successors of n, or $\mathbf{lg}(\mathcal{CG}, n) = \mathbf{sg}(\mathcal{CG} \cup \mathbf{a}^*(\mathcal{CG},l))$, where, l is a leaf and regular successor of n.[]

In Figure 4, part (a) and (b) are two study graphs for {UMR} and {UMR, QSR} of the curriculum graph shown in Figure 3. Part (c) and (d) are the learning graphs of leaf QMR and interior node SR, respectively.

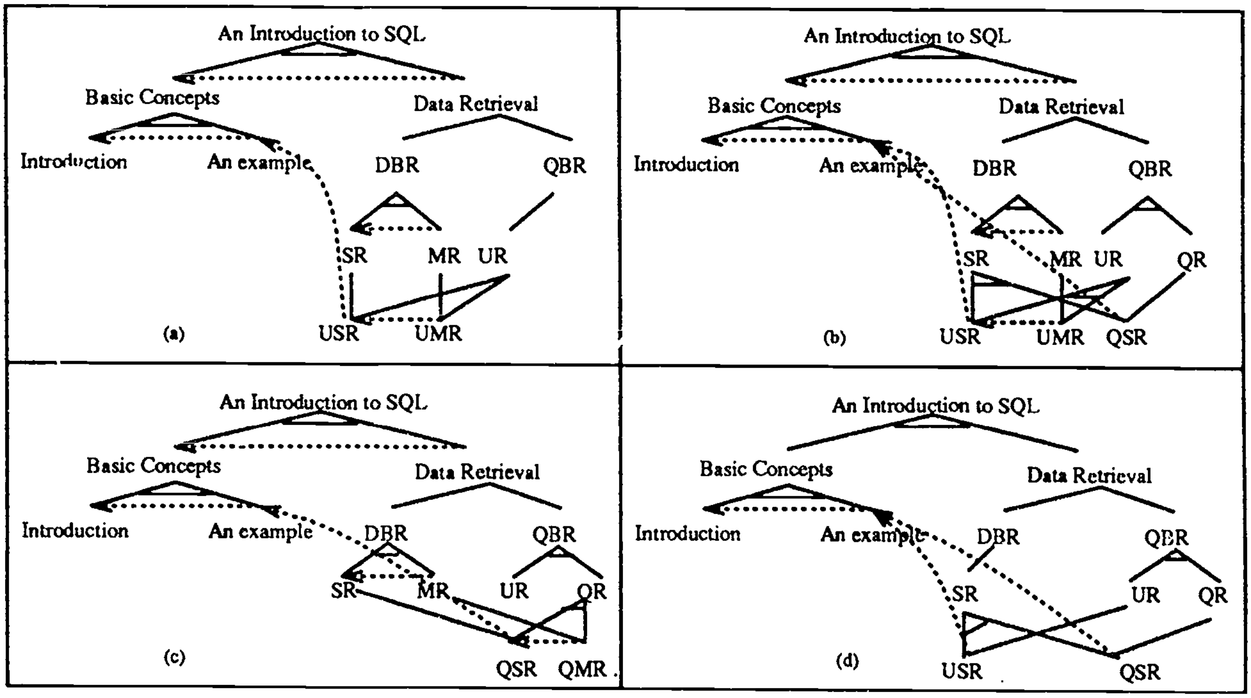


Figure 4: Examples of study graph and learning graph

Theorem 1 *A learning graph is always a well defined study graph.*[]

PROOF This is obvious from the definition, because for any leaf of a learning graph, all of its prerequisites are also there.

Given a curriculum graph $\mathcal{CG}$ and a m-node of $\mathcal{CG}$, if we only focus on one viewpoint and discard all others, do we still have a well defined curriculum? For the same graph, if we remove some node from it, how can we keep the remaining part well defined? The following theorems will answer these questions.

Theorem 2 *If $\mathcal{CG}=(G, M, GE, ME, P, R)$ is a curriculum graph, $m \in M$ is a m-node, and $n_i \in (M \cup G)$ is a child of m, then if we remove all the nodes which are regular successors of m, but not regular successors of n_i, from $\mathcal{CG}$, the result is still a well defined study graph.*[]

PROOF Omitted.

Theorem 3 *If $\mathcal{CG}=(G, M, GE, ME, P, R)$ is a curriculum graph, L is the set of its leaves, and $n \in L$ is a leaf, then $\mathbf{sg}(\mathcal{CG}, L-(\{n\} \cup \mathbf{s}^*(\mathcal{CG},n)))$ is a set of well defined study graphs.*[]

PROOF Omitted.

Corollary 1 *If $\mathcal{CG}=(G, M, GE, ME, P, R)$ is a curriculum graph, L is the set of leaves, and $N_s = \{n_1, n_2, \ldots, n_k\} \subset L$ is a subset of leaves, then $\mathrm{sg}(\mathcal{CG}, L-(N_s \cup \mathbf{s}^*(\mathcal{CG},n_1) \cup \ldots \cup \mathbf{s}^*(\mathcal{CG},n_k)))$ is a set of well defined study graphs.*

PROOF Omitted.

Definition 12 Let $\mathcal{CG}=(G, M, GE, ME, P, R)$ be a curriculum graph and $n \in (G \cup M)$ be a node. n is *monous* if all of its regular ancestors are g-nodes. n is *single parent* if it only has one regular incoming edge. []

Theorem 4 *If $\mathcal{CG}=(G, M, GE, ME, P, R)$ is a curriculum graph and $n \in (G \cup M)$ is a non-monous interior node, then the removal of n and all its single parent successors from $\mathcal{CG}$ results in a well defined study graph.*[]

PROOF Omitted.

An Object-Oriented Implementation

We have implemented a curriculum management system called **CM** (Curriculum Manager) by Common Lisp Object System (CLOS)[Keene, 1989]. Four types of classes have been defined in **CM**: **node**: the foundation of all nodes in a curriculum graph; **goal-node**: whose instances are g-nodes; **multi-view-node**: whose instances are m-nodes; **unit**: whose instances are units.

The system interface to a student is defined by a group of system commands: **display, focus, study, not-study, and skip**. These commands provide tools for users to construct course curriculum according to their individual requirements before the selecting period. The purpose of **focus** is to allow a student to focus on one view point while ignoring others on a goal. A student can use **study** to a select the specific topic he wants to study. The result is a new curriculum graph (study graph) with all the prerequisite nodes remained in positions but unrelated topics removed. On the other hand, he can use **not-study** to exclude whatever optional topic he is not interested in, while keeping the remaining curriculum well defined (in the sense that the result is a well defined study graph). The last operator is **skip**, is used to allow students to skip any part of a course. It differs from **not-study** in that the resulting study graph is not necessarily well defined. When students do not want to study some topic (because they have already mastered it), they can use **skip** to skip it; on the other hand, even if the students have not mastered some topic, they still can get rid of it by **not-study** (one possible reason is they have no interest in it).

In comparison with the existing KBTSs, our system has the following three major advantages:

- The curriculum knowledge about a course is represented explicitly in the system and the CKB is built up based on a theory of curriculum knowledge representation;
- The structure of a course could be tailored and multiple curriculum could be generated according to the special interests from students, thus each student can chose an individualized course from one system;
- Student can study the same part of the domain knowledge from different conceptual perspectives, and combined with the above feature, this offers well-prepared course material in lessons that are optimized for each student.

References

[Chan, 1992] Chan, Tak-Wai. (1992). Curriculum Tree: A Knowledge-Based Architecture for Intelligent Tutoring Systems. *Intelligent Tutoring Systems: Second International Conference on Intelligent Tutoring Systems, ITS'92.* Spring-Verlag.

[Lesgold, 1987] Lesgold, A. (1987). Toward a Theory of Curriculum for Use in Designing Intelligent Instructional Systems. *Learning Issues for Intelligent Tutoring Systems.* Springer-Verlag, New York.

[Keene, 1989] Keene, S.E. (1989) *Object-oriented Programming in Common LISP: a programmer's guide to CLOS.* Addison-Wesley.

[Wenger 1987] Wenger, E. (1987) *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge.* Los Altos, California: Morgan Kaufmann Publishers, Inc.

PANELS

PANEL

The use of hypermedia in the teaching and learning of programming

The Use of Hypermedia in the Teaching and Learning of Programming

TOM BOYLE
Department of Computing
Manchester Metropolitan University
Chester Street, Manchester M1 5GD, England

Learning to program involves a complex range of abilities in the design, coding, and testing of software. Students have to acquire skills in expressing solutions in a programming language, visualising the run time behaviour of their programs, and constructing tests of correctness. Increasingly, for many institutions, these skills have to be taught to large numbers of students from a wide diversity of backgrounds. A number of sophisticated learning environments have been developed to meet this teaching and learning challenge. These approaches differ in their emphasis on pedagogical approach and the areas of programming skill covered. Some of the systems are in mainstream use by thousands of students. Others represent advanced hypermedia prototypes. In this panel we will review what has been achieved in the use of hypermedia for the teaching and learning of programming and consider the design features of the next generation of hypermedia-based systems for programming education.

The panel members will describe their experience in designing a variety of hypermedia or hypermodal learning environments for programming. The DESIGN TOOL (University of Texas) and the CLEM and Braque systems (Manchester Metropolitan University) provide advanced interactive environments for learning design and implementation skills. The Genie systems (Carnegie Mellon University) provide sophisticated support for multiple program views and an integrated visual debugger to support hypermedia literate programming. The ISIS-Tutor (ICSTI Moscow) aims to link ITS and hypermedia components to provide adaptive guidance for the learner. The problems of developing hypermedia learning environments for large distance learning communities are being tackled at the Open University in England.

Based on their extensive experience the panel members will discuss the use of hypermedia features to support various aspects of the programming process. These include:

- the learning of abstract design skills
- learning language syntax and semantics
- support for program comprehension, visualisation and testing

I will raise a number of points based on experience with the CLEM and Braque systems. The main issues are:

1. the need for a coherent base in the psychology of learning and problem solving for the design principles which guide the construction of such systems. The psychology of natural language development has proved to be a particular rich source of insights;

2. how to maintain coherence in very large learning environments used by a wide variety of learners;

3. the role of empirical feedback on systems, based on use in real learning contexts, for both formative and summative evaluation;

4. the potential for the integration of design ideas and tools developed at different international centres.

The problems and proposed solutions discussed by the panel should be of interest to those involved in using hypermedia to support the learning of complex, structured skills.

Integrating Intelligence and Hypermedia in the Teaching and Learning of Programming

PETER BRUSILOVSKY
International Centre for Scientific and Technical information
Kuusinen Str. 21b, Moscow 125252, Russia

I will discuss our experience of designing a hypermedia component in an intelligent learning environment (ILE) for programming. An important feature of any ILE is the use of knowledge about the domain and the student. In our ITEM/IP and ISIS-Tutor environments we represent the domain knowledge in the form of a network of programming concepts and constructs. This network reflects the structure of the domain and is used as the basis for both the overlay student model and the hypermedia network, which represents all teaching material: explanations, examples, tests and problems.

One of the main goals of our ITEM/IP and ISIS-Tutor projects was to investigate the ways in which hypermedia components can be integrated with an educational programming environment and an intelligent tutoring system for programming. I will discuss the following three aspects of our approach:

1. Support of multiple navigation paths between hypernodes of different kind.

Each of programming concepts or constructs has a hypernode as external representation. Each teaching operation of any kind (examples, problems, etc.) is also represented as a hypernode. Each relationship between concepts is represented as a bi-directional pathway. Moreover, each teaching operation has bi-directional hyperlinks to all concepts and constructs related with this operation. At any concept hypernode the student can get lists of related concepts separately for each kind of relationship, lists of all related teaching operations separately for each kind of operation and navigate to any listed hypernode. At any teaching operation hypernode the student can get the list of all related concepts and constructs and navigate to it.

2. Integration of exploratory environment and hypermedia

Each teaching operation which deals with programs or programming constructs (examples, tests, problems) is a door to the programming laboratory. The laboratory is equipped with visual interpreter and structure editor and used to play with examples, to answer tests, or to solve problems. The student can also work with the laboratory in the exploratory mode.

3. Integration of guided tutoring and student-driven learning

The tutoring component and the hypermedia component of ILE use the same student model to represent the results of student learning. Using this model the tutoring component can suggest at any moment of learning next "best" teaching operation, thus providing the required guidance. If the student is not satisfied with system choice, she can select next teaching operation herself with hypermedia links.

4. Adaptive hypermedia.

The hypermedia component uses the student model to adapt itself to the given student. First, the content of the hypermedia page is adapted to the student knowledge. Second all the lists of hyperlinks to related concepts and teaching operations are adaptive. Not ready to be learned concepts and operations are dimmed and others are marked in two or three colors according to the current state of student knowledge on it. Visual marks support student orientation and navigation in the hyperspace of teaching material reporting what is new, what have been learned, or what needs further work.

The Use of Hypermedia in the Teaching and Learning of Programming

Philip Miller

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania U.S.A
plm@cs.cmu.edu

My remarks will be based on experiences gained in using the MacGnome Programming Environments. Included are the Karel Genie, the Pascal Genie, the Object Pascal Genie and the ACSE environment. These environments have been used in teaching programming and computational. We believe over 100,000 students have used one or more of these environments as a major programming environment in regular coursework. They have been used at a number of leading universities, colleges, and secondary institutions.

The Genies, as the MacGnome environments are collectively known, illustrate the power of multiple program views, complete incremental static semantics, a fully integrated visual debugger, a continual cycle of refinement based on user testing, and the full integration of hypermedia in support of hypermedia literate programming.

I will raise 5 issues:

1. What is the importance of the various system components? What can be eliminated from systems that actually do the job?

2. What is the impact of hypermedia as we have used it, on the learning process?

3. What role exists for hypermedia that we have not yet explored?

4. What does the distinction between teaching environment and professional environment imply?

5. What is our view of programming as an independent skill and how does that effect the tools we build and the way we teach?

An Environment Supporting Computer Science Education Integrating Hypertext, Program Design, and Language Facilities

WENDY A. LAWRENCE FOWLER
Department of Mathematics and Computer Science
University of Texas - Pan American, Edinburg, TX 78539, USA
fowler@panam1.panam.edu

I will discuss our experience integrating a visually-based design and programming environment with a hypermedia presentation of course content. The environment provides students with a suite of pedagogically oriented programs. The programs and interrelationships among programs are designed to support independent development of each of the skills necessary for programming, while keeping the integration of the skills and the more abstract concepts of computer science in sight.

A student oriented CASE tool supports a pedagogy which focuses on developing general problem solving skills as an integral part of programming. Design Tool uses a visually based system for problem decomposition, program design via structure charts, data flow checking, Pascal code generation, and report writing. Students' first experiences in the laboratory environment involve the completion of design exercises.

Design Tool provides general tree display facilities for module creation and deletion, module reordering and repositioning, and the collapsing and expanding of subtrees and module windows. An overview diagram provides an orienting view of the entire design and navigation facilities for moving around within the design. A student oriented pseudocode language, focusing on data flow and consistency, allows students to focus on algorithm development. Constraint checking eliminates errors in data flow prior to code generation and a move to the implementation facilities. Code generation facilities allow the student to design moderately complex problem solutions and remain focused on the problem solution and design, rather than on a particular language or programming environment.

A hypertext notebook serves to integrate concepts introduced in the classroom and text with details of design and implementation. The notebook contains the course lecture material, course syllabus, schedule, laboratory exercises, programming assignments, and a large set of designs and example programs. One feature of the hypertext, Quick Look screens, provide a brief explication of a concept. The screens provide a readily available and easy to use help facility for programming language semantics and data structure concepts analogous to the syntax help facilities in most programming environments. The hypertext is implemented in a system which allows direct transfer of information to both Design Tool and the programming environment, serving to physically and conceptually relate more abstract concepts to details of design and implementation

We have used questionnaires to assess student response to the tools, and attitudes toward design and implementation concepts. Pre- and post-test exercises, and a standardized series of assignments and exams provide insights as to the effect of the tools on student understanding and skill development in problem solving, design, and implementation in a structured programming language.

The discussion will focus on 1) how to integrate the principles of program design with the practice of programming through a student-oriented CASE environment, 2) how to broaden the scope of learning during the programming process through an integrated hypermedia presentation of information, and 3) the effect on students of the approach we have taken.

The Use of Hypermedia in the Teaching and Learning of Programming (A Distance Education Perspective)

MARK WOODMAN
Computing Department, Faculty of Mathematics and Computing
The Open University, Walton Hall, Milton Keynes MK7 6AA
United Kingdom
Email ids: m.woodman@open.ac.uk

Hypermedia technology is of significant potential benefit to students in the distance learning mode – isolated from their peers and tutors. However, there exist significant problems to do with the design and use of such hypermedia systems to support the teaching and learning of programming.

I will report on current developments in multi-media teaching of programming and software development at the Open University and discuss issues which are of strategic importance for both distance learning and for student-centered learning in the conventional sector. The Open University is the UK's distance learning university, offering an open-access modular degree programme which includes high-population courses in computing. The average age of student is 35. At present most students in the introductory course have previously studied a foundation course in mathematics or technology. Despite this fact we are observing a decrease in abstraction skills and a reduction in the ability to manipulate symbols. Over the past five years we have increased our number of students studying courses in software development to more than 6,500 per year. I will report on the use of hypermedia in a major project to replace our flagship introductory course at the beginning of 1997; the new course is planned to have in excess of 3,500 students per year.

A number of problems face the team developing the course and the solutions to these may lie in the use of advanced multi-media technologies. I will describe how these have been addressed in the functional specification and design of software for the new course. We are developing a hypermedia-based teaching environment to which our own and third-party software must interface.

In particular the following issues will be addressed:

1. Concepts of programming and software development are not being sufficiently deeply understood by students. To some extent we believe that current approaches to the teaching of programming do not develop abilities to reason about complex software and are developing tools to assist students in this respect. How can we use hypermedia to improve abstraction skills and deepen understanding of programming?

2. Increasingly students start software courses with little experience of manipulating symbols and many will have no experience of independent study. To what extent can a lack of mathematical, problem solving and notation manipulation skills be mitigated using hypermedia? In addition, how can the technology simultaneously cater for those who have few initial skills and those who are impatient to progress?

3. The cost of changing printed correspondence material and recording audio and visual material for broadcast is high. Do we need to organize our media differently if it is to be computer-based? Strategies for mixing computer-based teaching, printed material, and television and radio broadcasts will be described; in particular, paper-based hypertext that mimics computer-based systems will be discussed.

4. Software and hardware technology changes at a rate that is impossible to ignore and difficult and costly to track. Will the multi-media systems we develop really be amenable to change or will they become too tightly coupled to software development methods and programming language types?

PANEL

Virtual Clayoquot video database: The Bayside Middle School implements a networked multimedia socio-scientific study about a British Columbia rainforest

VIRTUAL CLAYOQUOT VIDEO DATABASE: The Bayside Middle School Implements a Networked Multimedia Socio-Scientific Study about a British Columbia Rainforest

RICKI GOLDMAN-SEGALL
MERLin, Faculty of Education, University of British Columbia
Vancouver, British Columbia, Canada V6T 1Z4
ricki@unixg.ubc.ca

PANEL MEMBERS:

KEVEN ELDER
Vice-Principal, Bayside Middle School
Brentwood Bay
Vancouver Island, BC
Keven_Elder@bays.etc.bc.ca

TED RIECKEN
LESLIE FRANCIS-PELTON
Faculty of Education
University of Victoria
Victoria, BC
triecken@postoffice.uvic.ca

JORDAN TINNEY
LAURI ROCHE
KATHRYN GODFREY
Teachers, Bayside Middle School
Brentwood Bay, Vancouver Isl., BC
Jordan_Tinney@bays.etc.bc.ca
Lauri_Roche@bays.etc.bc.ca
Kathryn_Godfrey@bays.etc.bc.ca

MIKE HOEBEL
Educational Technology Center
Ministry of Education, British Columbia
PO Box 2040, Sidney, BC V8L 3Y3
mhoebel@cln.etc.bc.ca

STUDENT/S (TBA)
Bayside Middle School

Introduction

Virtual Clayoquot is the name given to a science and social studies computer-based learning environment for teenage girls and boys from the Bayside Middle School, a Vancouver Island junior high school, to explore the controversy over the future of a rain forest, on the west coast of Vancouver Island, called Clayoquot Sound (KLAK-WIT Sound). Over the coming years, we expect to (1) find out more about how these teenage girls and boys make sense of complex scientific issues – when the issues are significant to their daily lives and when they are studied across the curriculum, and 2) to find out how using networked multimedia applications will build a distributed community of inquiry. The collaborative team represented in this panel at EdMedia'94 consists of students, teachers, professors from two British Columbia universities, industrial liaison members, community-based science specialists and educators, and representatives from the Ministry of Education. Produced over the next few years will be the product, *The Virtual Clayoquot Magazine*, a series of CD-ROM disks. The purpose of the *Virtual Clayoquot* project is to enable young people to conduct their own research in order to become informed decision-makers about socially important scientific issues. Our plan is to have these young people electronically discussing their research with other children from rainforest and wetland regions, using a mail program called *CLIENT*, developed at the Educational Technology Center (ETC). From the moment the young people began their investigations, a team of faculty and graduate student researchers initiated an ethnographic research about the thinking styles used by these young people, paying particular attention to the different approaches used by teenage girls and boys. Thus, issues concerning gender and science (Linn & Hyde, 1989) play a significant role in both the content and style of the video and text data that is being collected and analyzed. A data analysis software application called *CONSTELLATIONS* (Goldman-Segall, 1994) is the tool being used for building theories about the children's thinking and will be used as the interface for accessing the *Virtual Clayoquot* data on the CD-ROMs.

Theoretical Premises

Our research is based on three premises. The first premise is that a relevant subject mat e as controversial as the forest dispute at Clayoquot Sound will provide powerful content across the curriculum area for this group of young people to construct their own transformative theories. Another underlying premise is that learning occurs best when people study in a manner that is complementary to their own preferred way of thinking (Goldman-Segall, 1991). (This does not imply that learning does not occur when the subject matter is not relevant or requires using a non-preferred style of thinking.) The third premise is that emerging electronic media can support human inquiry by providing ethnographic methods of evoking and provoking the learning process. In fact, multimedia learning, in this environment, becomes a post-modern ethnographic encounter wherein the researcher, the researched (the content), and the reader of the research negotiate meaning. As Tyler points out:

> A post-modern ethnography is a cooperatively evolved text consisting of fragments of discourse intended to evoke in the minds of both reader and writer an emergent fantasy of a possible world of commonsense reality, and thus to provoke an aesthetic integration that will have a therapeutic effect. (Tyler, 1986, p. 125)

Learning in a constructionist multimedia environment is also an aesthetic integration that blends the learner, the various media (including the teacher), and the message (or content) into one contiguous web.

The second premise requires explanation as it deals with how we know what we know about the world around us, our epistemological point of view. Traditionally, epistemological theories support the existence of levels or stages of knowledge – Piaget's stages of intellectual development being a good example. Carol Gilligan (1982) suggests that there is a "different voice" or model to the hierarchical one. Turkle and Papert (1991) call for an epistemological pluralism, where both "hard thinking" and "soft thinking" are equally significant styles of thinking. Papert's learning theory recognizes all learners as constructionists moving from novice to expert in accordance with their style of thinking. Papert and Turkle's theory is that, traditionally, hard thinking in science has been the basis of defining logical thinking. "And logical thinking has been given a privileged status which can be challenged only by developing a respectful understanding of other styles..." (1991). According to Papert & Turkle (1991), soft thinking is more negotiational and contextual, or what Lévi-Strauss terms, *bricolage*. In other words, science is not the pure, "objectively" neutral endeavor that traditional theorists once claimed it was. More and more, we are coming to agree that science is a culturally shaped construct by those who engage in its activities (Solomon, 1987). As the people who do science change, the paradigms change (Kuhn, 1970). A relational approach to conducting scientific explorations offers a wider range of ways to doing science. Fox Keller calls science a "deeply personal as well as a social activity" (1985, p.7). In the *Virtual Clayoquot* study, the relational and personal approach to doing science is expanded upon in order to closely examine epistemological pluralism. We might hypothesize that scientific knowledge is also a tool for relationships with the world in which we live, and not only an area of inquiry in and of itself. Thus, studying the Clayoquot provides an opportunity for students to use their relational, narrative, or empirical styles of thinking to build theories about a complex subject.

Method of Conducting *Virtual Clayoquot* Study

Young people are partaking in three diverse roles in this project: they are researchers, designers, and distance educators.

Young People as Researchers: Several teams of young people and their teachers are investigating the issues surrounding the dispute primarily concerning the future of old growth forests and the economic well-being of local communities. Teenage girls and boys have opportunities: to visit the site; videotape their impressions of the forests; interview loggers, protesters, community members, and government officials; and collect relevant articles from newspapers, magazines, and scientific journals. Along with adult expert resource persons, they interpret the video and text data from a variety of perspectives: forest and wildlife issues; employment needs; First Nations peoples' claims; municipal and provincial priorities; trade agreements; and, global concerns , to name a few.

Young People as Science Curriculum Designers: Girls and boys of the Bayside Middle School will use *CONSTELLATIONS* to classify, categorize, link, and analyze the data. Students will search through large databases, build "constellations" or groups of data, and construct interpretations based on their own findings. They will selecting the most relevant data as the basis for a learning module to be made available on CD-ROM.

Young People as Distance Educators: We expect that this multimedia module on Clayoquot Sound will be composed so that other young people will be encouraged to make informed decisions about this controversial social, cultural, and scientific issue. Our long-term vision is to take advantage of emerging technologies such as large networks that can encode and decode multimedia information. This will enable young persons from the

Bayside to have real-time video-based discussions with children in other schools. Currently, these young people are in electronically linked to the World Wide Network using *CLIENT*.

The following is a list of the activities that the young people are either engaged in currently, or will be engaged in within the following years of the project:

- investigating the issues surrounding the Clayoquot Sound dispute from as many perspectives as possible and through as many recorded media as available.
- debating the issues in class on a semi-regular basis and have this be part of the database.
- visiting the site and conduct a series of interviews using video, tape, pen and paper, and portable computer.
- assembling the text, sound and video data to analyze using *CONSTELLATIONS*.
- designing the overall CD-ROM architecture for other young people.
- conducting school surveys about attitudes and ideas for change.
- using human expert resources at various locations in British Columbia and elsewhere.
- becoming resource persons for others interested in the rain forest issue in BC.

The following is a list of the activities that the faculty and graduate student researchers are either engages in currently, or will be engaged in within the following years of the project:

- involving young people in our research about them and let them study us, if they choose.
- co-ordinating the methodological approach for analyzing the data about the process.
- conducting interviews and observations of events that demonstrate the specific ways in which gender differences can enrich the scientific process.
- developing a more relational approach to reporting our findings.
- becoming part of the student research team so that we are all making contributions that bring out our best.
- studying and share what we know and what we don't know with our peers and our community.

Multimedia Tool, *CONSTELLATIONS*, for Analyzing Data

Over the past eight years, I have been working closely with colleagues from the MIT Media Lab and with computer scientists at UBC. During this time, I have been developing a theory of how to layer data so that the meaning of the a recorded event could be easily communicated and understood in a multimedia documentation (Goldman-Segall, 1993). I also designed with two separate research teams different versions of *CONSTELLATIONS*, which are software applications for annotating, linking, and juxtaposing media forms into multimedia documents. In both systems, the researcher can build layered descriptions upon which she and others can e-VALUE-ate the validity of her conclusions in the following ways:

- Building groupings that are similar (clustering, stacking)
- Thickening the description with annotations
- Linking across attributes: slicing through the layers
- Signifying the meaning by add "weight" to the attributes
- Adding perspectives and "points of viewing;" triangulation
- Juxtaposing video/text/sound in diverse configurations
- Making fine-grained selections
- Fine tuning, trimming, narrowing focus, reaching core concepts

Let me expand upon a few of these ways of layering and explain how students and researchers could use this method for their investigations. One way of layering our understanding occurs during the process of interpretation when multiple users comment upon, or annotate, the same video stream. As they annotate, they thicken (Geertz, 1973) the data because each user interprets the same event quite differently. Multimedia designers are now beginning to realize that diverse users manipulate video data differently. The likelihood of conclusions being the same is minuscule. The possibility of them falling into the same range (Geertz, 1973) is likely.

Having access to multiple forms of representations of research data – video, text, sound – also adds to the building of layers. The days where text was the only legitimate form of data are over. The use of sound, photographs, video or film, or drawings and sketches have become commonplace. However, most computer applications treat researchers as being married to one form — usually to text. It is possible to add a digitized image or a QuickTime movie to a text document, but if the researcher wants to codify this data with the other data, she must resort to using either a text-based, video-based, or sound-based application. In *CONSTELLATIONS* all data forms can be codified and linked as equal but different forms.

However, the kind of layering that is more crucial is a result of what happens when data are shared. How can members of a research team explore the various significant ratings of colleagues? In other words, can layering enable the researcher to get a sense of seeing the data from other perspectives in order to triangulate interpretations? These and other questions will be addressed in this panel discussion and demonstation of the tools we are using.

Panel Topics

Instead of concluding with the findings we have made to date, I would like to take this opportunity to highlight a few of the central topics that the team of panelists will be presenting as the first phase of the project. draws to a close Jordan Tinney, a teacher at the Bayside, will describe the process of enabling his students to study the various points of view surrounding the dispute. Tinney's students (grades 6 and 8) studied the Clayoquot Sound dispute for over two months in his social studies classes. He will describe the projects students assembled and the kind of thinking they had about the issues before and after the immersion. Lauri Roche is the teacher of a special needs class. Three of her students participated in Jordan's class throughout this first stage. Roche will describe the creative approaches used by her students to offer solutions for how to resolve some of the issues. Kathryn Godfrey has been active in global education for several years. Her class has taken a broad view of how to understand ecological issues. She will describe several activities that her students have done to help them reflect upon these issues. Keven Elder, the vice-principal of the school, will provide the overall school commitment to exploring multiple points of view and will explain the philosophy of the school culure in light of new technologies, as it approaches the year 2000. Ted Riecken, a faculty researcher from the University of Victoria, will discuss the overall research agenda at Bayside and how this project fits into a pattern of research that he and his colleagues initiated three years ago when the Bayside teachers were still at the old school site. Rieken will also describe his methodological approach in building narrative ethnographic accounts. Leslee Francis Pelton, Riecken's colleague at UVic, will describe how an issue such as Clayoquot can be studied across the curriculum.. She will bring her mathematical perspective to the project showing how issues such as Clayoquot Sound help children to understand the interaction of Science/Math, Technology and Society. Michael Hoebel from the Educational Technology Center (ETC) will discuss network issues with other schools and the BC commitment to learning technologies. In fact, Hoebel and his team at ETC, have been the technology pioneers in British Columbia. Hoebel will be able to describe, not only the current work he is involved in at the Bayside, i.e., *CLIENT*, but provide the historical perspective of how computers have entered into the lives of children and their teachers in BC schools. Goldman-Segall will show data that has been gathered in phase one, show excepts of the data, report findings to date, and describe the next phase of this action-oriented research project. We welcome you to join us to hear about the developments and contribute to its progress through your questions and quieries.

This research is supported by a Social Sciences and Humanities Research Council of Canada (SSHRC) grant called, Video and Validity: A Multimedia Collaborative Research Tool for Representing What We Record, Interpret, and Share with Others and by the Natural Science and Engineering Research Council (NSERC) of Canada Strategic Research Grant (#5 81457) called Logging, Annotation and Navigation for HyperMedia Video Analysis Tools.

References

Fox Keller, E. (1985). *Reflections on Gender and Science.* New Haven and London: Yale University Press.

Geertz, C. (1973). *The Interpretation of Cultures*, Basic Books: New York.

Goldman-Segall, R. (1994). *CONSTELLATIONS.* Multimedia Software. Vancouver, BC: MERLin Lab, Faculty of Education, UBC.

Goldman-Segall, R. (1993). Interpreting Video Data. *Journal for Educational Multimedia and Hypermedia.* Association for the Advancement of Computing in Education, 2, (3).

Goldman-Segall, R. (1991). Three children; three styles: A call for opening the curriculum. In I. Harel & S. Papert (Eds.), *Constructionism.* Norwood, New Jersey: Ablex Publishers.

Gilligan, C. (1982). *In a Different Voice: Psychological Theory and Women's Development* Cambridge, MA: Harvard University Press.

Kuhn, T. (1970). *The Structure of Scientific Revolutions.* 2nd ed. Chicago: University of Chicago Press.

Linn, M. C. & Hyde, J.S. (1989). Gender, mathematics, and science. *Educational Researcher,* 18, (8).

Motherwell, L. (1988). *Gender and Style Differences in a Logo-Based Environment.* Unpublished Doctoral Dissertation, Cambridge, MA: MIT Media Laboratory.

Solomon, J. (1987). Social influences on the construction of pupils' understanding of science. *Studies in Science Education.* 14.

Turkle, S. & Papert, S. (1991) Epistemological Pluralism. In I. Harel & S. Papert (Eds.), *Constructionism.* Norwood, New Jersey: Ablex Publishers.

Tyler, S. (1986). Post-Modern Ethnography: From document of the occult to occult document. In J. Clifford & G. Marcus (Eds.),*Writing Culture.* Berkeley: University of California Press.

PANEL

Internetworking for K-12 education

Internetworking for K-12 Education

VICKI L. HANSON
IBM Thomas J. Watson Research Center
P. O. Box 704, Yorktown Heights, NY 10598

The use of the Internet in K-12 education has the potential for radically changing the way students learn. This resource represents thousands of connected networks around the world that allow teachers and students to access and share resources beyond their school walls. It provides a unique ability to collaborate and communicate with other students and teachers world-wide. It also provides a timeliness for current events and data-gathering otherwise unavailable.

Simply giving access to this rich array of information, however, does not guarantee its effective use. If schools are to be more than passive receivers of this information, they will need to learn how to productively explore and use this resource. Spurred by interest and funding at the state and national levels, researchers have begun investigating issues related to training of teachers in the use of this technology and related to the exploration of new instructional techniques using collaborations of students from different parts of the globe.

Participants in this panel have all been involved in pioneering projects designed to make the Internet accessible for both students and their teachers. Panelists will discuss their large-scale state and national pilot projects on teacher training, classroom reform, and educational policy related to introduction of the Internet into K-12 education.

In Texas, for example, teachers and students use a state educational network to link to Internet resources. The number of teachers using this system has grown exponentially, indicating educators' interest in taking advantage of these resources. Their use of the Internet through this state system will be discussed.

The Kids as Global Scientists project deals with questions of classroom reform, centered around the use of the Internet. The results of this research suggest ways in which the type of knowledge students acquire through this collaborative learning experience differ from the type of knowledge they acquire through traditional instructional approaches.

The NSF-National School Network Testbed has been addressing technical and policy issues involved in scaling up the Internet to a point where it is universally accessible by all K-12 students, teachers, and administrators. In large part, this involves making available information, materials and services of sufficient value to generate investment by schools and government.

New Learning and New Learning Models with Kids as Global Scientists

NANCY BUTLER SONGER
School of Education and Institute of Cognitive Science
Campus Box 249
University of Colorado, Boulder, Colorado , U.S.A. 80309

The Kids as Global Scientists (KGS) project focuses on the orchestration of classroom reform using innovative learning models and telecommunications technologies to bring the excitement of "real scientists'" investigations into middle school classrooms. In addition, we couple our design of reform classrooms with detailed investigations of student learning.

Students in our Internet Hub classrooms utilize a Hypercard front-end software tool called *InternetTrek* to regularly access real or near-time spectacular satellite images and weather maps which characterize the path of weather phenomena over the globe. Simultaneously, they correspond through electronic coordinated exchanges with middle school students and professionals world-wide. Most importantly, the coordinated activities and software support students' investigation into their own questions in Atmospheric Science. Our activities promote questioning, collaboration and the exchange of first-hand information from students who are experiencing the phenomena and who might be thousands of miles from their correspondents' classrooms. In addition, the anonymity of electronic correspondence has, in several instances, encouraged students to excel who are sometimes marginalized by face-to-face peer activities.

Preliminary results indicate that students using our Global Exchange learning model and telecommunications technologies for the gathering and exchange of information demonstrate distinct differences in the type and value of knowledge developed from what might be obtained in more traditional learning environments. We have found that learning about science as the science is occurring (which we call *today's knowledge*) and through dialogue and exchange with peers who share similar perspectives (which we call *interactive knowledge*) are powerful approaches to learning. We have begun to characterize these emerging learning differences, and their influence on student knowledge development and motivation to learn science. To illustrate one aspect of today's' knowledge, we share a quote from a student, "They [the students in the other locations] give you their point of view of the weather not what other books give you...every day is the same to books." We continue to investigate the nature of the knowledge developed, and the potential for our classroom reform to enhance understandings.

PANEL

Can electronic games make a positive contribution to the learning of mathematics and science in intermediate classrooms?

The Educational Potential of Electronic Games and the E-GEMS Project

M. M. KLAWE
Department of Computer Science
University of British Columbia, 2366 Main Mall, Vancouver, B.C.,V6T 1Z4, Canada
email: klawe@cs.ubc.ca

The panelists are members of the Electronic Games for Education in Math and Science (E-GEMS) research team. E-GEMS is a collaborative effort by educators, computer scientists, electronic game designers, scientists, curriculum designers, and mathematicians, to increase understanding of how math and science learning can be aided by the use of electronic games. E-GEMS is engaged in the study of existing electronic games and in the design of new games in an effort to encourage more children to explore issues in mathematics and science, combining learning through electronic games with school curricula, tools, materials, and practices.

Electronic games are an important part of the popular culture of many North American children. Video game arcades have been eclipsed by the home video game units found in countless living rooms -- *Nintendo* has become a household word. Even children who don't have a video game system at home are aware of this sweeping phenomenon: video games are popular and influential (Papert, 1993; Provenzo, 1991).

Few schools have incorporated video game technology into the curriculum. If electronic games are played in school settings, they are most often played on computers, and the games have ostensibly been designed for educational purposes. It is unlikely that the designers of commercial video games -- like *Street Fighter II* -- had explicit educational goals in mind. Nevertheless, the panelists hold the view that electronic games -- both computer and video -- have a place in the school curriculum, and moreover, that electronic games can be designed to enhance the teaching of mathematics and science in intermediate (Grades 4-8) classrooms.

I will open the panel by taking the position that electronic games can do more than entertain children. I believe there could be considerable curriculum content embedded in many electronic games, even in games that appear to be primarily designed for entertainment. The exploratory and interactive nature of electronic games is ideally suited to encouraging the exploration of mathematical and scientific concepts. Moreover, electronic environments facilitate visualizations and manipulations that are difficult to achieve with concrete materials. Existing games that could be used in Grade 4-8 classrooms as an effective aid to teaching math and science include *The Incredible Machine* for physics, *SimCity* for many mathematical concepts including percentages, rates, measurement, perimeter and dynamical systems, and *Lemmings* for logic, percentages, and programming. Many popular video games (e.g., *Super Mario World, Zelda, Secret of Mana*) contain excellent puzzles for motivating the study of patterns and structures. E-GEMS has developed *Monkey Math*, a collection of prototype games and activities for the exploration of negative numbers and fractions.

Future research includes the possibility of using the attractiveness of video and computer games to challenge children to take interest in math and science. I believe electronic games can enhance the interest and confidence of girls with respect to math and science, interest and confidence which often is lost in the intermediate grades (Linn & Hvde, 1989; Woods & Hammersley, 1993). Also, I am convinced that electronic games can be used to encourage boys who are likely to drop out of high school, by capitalizing on their interest in electronic games, and using those games to help them become interested in school curricula.

References

Klawe, M.M. (1992). Bringing Mathematical Research to Life in the Schools, *Proceedings of the 7th ICME Conference*, Quebec, 1992; Department of Computer Science, University of British Columbia, Technical Report, 92-33.

Linn, M., & Hyde, J. (1989). Gender, Mathematics, and Science. *Educational Researcher, 18* (8), p. 17-27.

Papert, S. (1993). *The Children's Machine: Rethinking School in the Age of the Computer.* New York: Basic Books.

Provenzo, E.F. (1991). *Video Kids: Making Sense of Nintendo.* Cambridge, MA: Harvard University Press.

Woods, P. & Hammersley, M. (Eds.)(1993). *Gender and Ethnicity in Schools.* NY: Routledge, Inc.

Teacher Mediation in an Electronic Games Environment

J. LAWRY
Department of Computer Science
University of British Columbia, 2366 Main Mall, Vancouver, B.C.,V6T 1Z4, Canada
email: lawry@cs.ubc.ca

I will comment on the types of learning that can occur through the playing of electronic games. I take the position that learning is more likely to occur through effective mediation on the part of teachers and/or teacher-researchers.

Imagine a "school" in the future, where children clamber to learn about long division, fractions, percentages, negative numbers, spatial relations, and the laws of physics by playing exciting electronic games. Gone are the repetitious worksheets, the tedious blackboard explanations by the weary teacher, and the uninterested looks on the faces of the students. Sound unbelievable? I think so, too, at least in part.

Electronic games are attractive to children, and learning can occur through the playing of electronic games. However, learning by playing is more likely to occur if it is coupled with effective mediation on the part of teachers. That is, to increase the realism in the above scenario, add a knowledgeable teacher who is comfortable with electronic games as teaching tools. Add, too, some related activities where the children explore, abstract, generalize, and extend the concepts they first became interested in by playing the game. Try to leave out the worksheets and the lectures.

Most children learn better when the goal concepts are embedded in a known, likable context. Electronic games can provide rich, attractive environments for initiating interest in math and science concepts. Once the interest is present, exploration of the game can be made more conducive to learning if the player is guided by an insightful mediator. A teacher can help the student to identify concepts in the game environment, as well as to abstract and to generalize the concepts by applying them in novel situations.

In our studies, we have found that many children interacting with a computer animation learn more about spatial concepts when the use of the computer is mediated by a teacher-researcher, and when the teacher-researcher uses manipulative materials to demonstrate and clarify the concepts contained in computer animation (Upitis et al., 1994). In addition, when children work in pairs, along with a teacher-researcher, they find the learning more enjoyable, use the computer for longer periods, and perform better on an external measure of learning (Inkpen et al., 1994).

We have also found the need to supplement an electronic game environment while using a game to teach children about gears and pulleys. Teacher-researchers guided play of the game *The Incredible Machine* to emphasize target concepts, used physical models to clarify the graphical simulation, and offered explanations that corrected inaccurately simulated physical laws. Despite the need for teacher augmentation -- or perhaps because of it -- children still enjoyed learning about gears and pulleys by playing *The Incredible Machine*.

By understanding the role of mediation, we will be in a better position to develop games and supporting materials that facilitate meaningful intervention by the teacher. Understanding and promoting the relationship between the teacher and educational materials is an essential step in the design of appealing electronic games that can be sustained in a school environment.

References

Inkpen, K., Booth, K., Klawe, M., & Upitis, R. (1994). *Cooperative Learning in the Classroom: The Importance of a Collaborative Environment for Computer-Based Education.* Department of Computer Science, University of British Columbia, Technical Report 94-5.

Upitis, R., Dearden, R., Inkpen, K., Lawry, J., Davidson, K., Klawe, M., Thorne, N., Hsu, D., Leroux, S., Sedighian, K., Anderson, A., Scharein, R. (1994). *Weird-a-gons and Other Folded Objects: The Influence of Computer Animation, Paper Models, and Cooperative Mediation on Spatial Understanding.* Department of Computer Science, University of British Columbia, Technical Report.

Gender Differences in an Electronic Games Environment

K. INKPEN
Department of Computer Science
University of British Columbia, 2366 Main Mall, Vancouver, B.C., V6T 1Z4, Canada
email: inkpen@cs.ubc.ca

Both boys and girls can learn about math and science through the playing of electronic games. However, I also recognize that there are gender differences in the ways that boys and girls approach electronic games environments.

Our studies indicate that girls enjoy playing electronic games, particularly computer games (Inkpen et al., 1993). Further, girls often prefer to play with a friend, and will play longer if a friend or teacher is taking an active interest in the game. When asked to design their own video games, girls tend to concern themselves with the personalities of the characters in the video games and on a story line. They seem less concerned with tools and logistics of game design. We need to keep these trends in mind if we are to capture the interest of girls through the electronic game medium.

Many boys are sophisticated game players, although this sophistication need not be at the expense of losing interest in other activities or in school. Boys are often attracted to the video game subculture -- they trade games and magazines, attend game-playing parties, and have many conversations about video games. They also appear to be attracted to the fast action and violence of video games, although our studies indicate that most boys will play cooperatively, or will develop sophisticated turn-taking systems to ensure that everyone gets a chance to play (Lawry et al., 1994). We do not endorse the violence in video games, nor the subservient role that female figures often play. But will we lose the interest of boys if we incorporate elements of cooperation and collaboration into electronic games, or create games that don't pitch good against evil? Are there ways to design games that attract both boys and girls, and contain substantive math and science content? I believe there are.

Gender differences are also apparent in the cooperative play of children in an electronic games environment. In a study using *The Incredible Machine*, children were observed playing the computer game in various cooperative settings (Inkpen et al., 1994). Girls appeared to perform better when a pair of children played together on one computer. Boys in the study were most successful when two boys played together, each using their own computer. Sharing of the mouse during cooperative play also yielded interesting results. Girls showed a higher proportion of non-contact requests and few requests involving physical touching of the mouse or their partner. In addition boys exhibited a larger number of refusals to pass over control of the mouse than girls. With the current emphasis on cooperative work in the schools, as well as the focus on cooperative work in the business world, it is important to recognize and understand the behaviours exhibited by both genders in cooperative environments. With that understanding, game producers can generate appropriate games and interfaces, which, when used with supporting curriculum materials, can increase the impact of electronic games in home and school environments.

References

Inkpen, K., Upitis, R., Klawe, M., Lawry, J., Anderson, A., Ndunda, M., Sedighian, K., Leroux, S., & Hsu, D. (1993). *"We Have Never Forgetful Flowers in Our Garden": Girls' Responses to Electronic Games.* Department of Computer Science, University of British Columbia, Technical Report 93-47.

Inkpen, K., Booth, K., Klawe, M., & Upitis, R. (1994). *Cooperative Learning in the Classroom: The Importance of a Collaborative Environment for Computer-Based Education.* Department of Computer Science, University of British Columbia, Technical Report 94-5.

Lawry, J., Upitis, R., Klawe, M., Anderson, A., Inkpen, K., Ndunda, M., Hsu, D., Leroux, S., & Sedighian, K. (1994). *Exploring Common Conceptions About Boys and Electronic Games.* Department of Computer Science, University of British Columbia, Technical Report 94-1.

Playing Styles for Computer and Video Games

K. SEDIGHIAN
Department of Computer Science
University of British Columbia, 2366 Main Mall, Vancouver, B.C., V6T 1Z4, Canada
email: kamran@cs.ubc.ca

One of the potential contributions of electronic games is that they can simulate a variety of different contexts since the adaptable nature of the technology allows children to explore phenomena that would be difficult to simulate with other media. Another attractive feature of the electronic medium is that many different styles of play can be accommodated. These include having children play alone or cooperatively, or contributing without actually playing games (e.g., standing aside and watching and/or making recommendations). Children playing games can use a variety of strategies to solve problems, including trial-and-error or logical-sequential strategies. I have observed many strategies and styles used in the playing of computer and video games.

I also hold the view that children's playing strategies and styles can closely reflect their learning and thinking styles. Research shows that people have a diverse set of learning preferences and styles (Keefe, 1987; Lall & Lall, 1983; Schmeck, 1988). Keefe (1987) states that learning styles are hypothetical constructs that reflect the underlying causes of learning behaviour. He proposes a learning style model that has three dimensions: cognitive, affective, and physiological. All these dimensions determine people's preferred modes of learning. Torrance and Rockenstein (1988) provide an overview of a number of these learning preferences and styles: some learners are more inclined towards a logical-sequential structure in which thinking and problem solving are performed in a step-by-step and data-driven fashion, some prefer to learn through discovery, and some learn things in an intuitive way and make accurate generalizations. In practice, however, people seem to mix and match a number of these styles, adapting their choice to the particular problem they are trying to solve or to the creation they are trying to construct. Others, however, cultivate only a narrow band of styles, and by doing so, may fail to realize the potential of alternate approaches.

I believe that electronic games can be used to augment and broaden the skills, strategies, and styles in a child's repertoire. For instance, a slow-paced puzzle-solving game may promote a logical-sequential style. An adventure game, where the player has an array of tools from which to choose in order to overcome obstacles, may help develop discovery-based learning in children. In addition, there is research indicating that fast action games may contribute to spatial visualization (Lowery & Knirk, 1982). Traditionally, schools have promoted and rewarded a limited set of learning styles (e.g., logical-sequential approaches to mathematics). Even teachers who successfully accommodate a divergent set of styles can benefit from having electronic games that appeal to children with differing learning and playing styles. For these reasons, I am especially interested in hybrid learning environments -- where a single electronic game provides the player with multiple learning-style possibilities; where children learn through social interaction with their peers, parents, and teacher(s); and where the school setting provides many learning activities such as games, manipulatives, instruction, and so on, to support the playing of games in school and home environments.

References

Keefe, J.W. (1987). *Learning Style: Theory and Practice.* Reston, Virginia: National Association of Secondary School Principals.

Lall, G.R., & Lall, B.M. (1983). *Ways Children Learn -- What Do Experts Say?* Springfield, Illinois: Charles C. Thomas Publishers.

Lowery, B.R. & Knirk, F.G. (1982). Micro-Computer Video Games and Spatial Visualization Acquisition. *Journal of Educational Technology Systems, 11* (2), 155-166.

Schmeck, R.R. (1988). *Learning Strategies and Learning Styles.* NY: Plenum Press.

Torrance, E.P. & Rockenstein, Z.L. (1988). Styles of Thinking and Creativity. In Schmeck, R.R. (Ed.). *Learning Strategies and Learning Styles.* NY: Plenum Press.

Parent and Teacher Attitudes towards Video and Computer Games

R. UPITIS
Faculty of Education
Queen's University, Kingston, Ontario, K7L 3N6, Canada
email: upitisr@qucdn.queensu.ca

The panel will close with an exploration of both favorable and critical beliefs held by teachers and parents about the playing of electronic games. While some parents and educators hold the view that electronic games can be valuable learning tools, many others believe that electronic games, especially video games, are at the very least vacuous and a waste of time, and at worst, lead to violent and aggressive behaviour. It would appear that at least some research substantiates the latter view (Graybill, 1985; Silvern & Williamson, 1987).

Although I enthusiastically share the goals of my colleagues on the E-GEMS research team, I myself fall into the class of adults who find video games unattractive. I have grave concerns about the violence embedded in many of the games, and about the ways that female figures are often depicted -- as passive creatures to be saved or prizes to be kissed (Provenzo, 1991). I also worry about children spending time playing video games when they could be doing other things -- playing outdoors or reading a book. These types of concerns are expressed by those parents and teachers who limit the amount of time the children in their care are permitted to play video and computer games, or ban games completely from the home and/or school (Lawry et al., 1994).

Nevertheless, I recognize the enormous potential of electronic games, and I realize that many children balance video game play with other activities and appear not to be adversely affected by the violent aspects of some of the games (Dominick, 1984; Lawry et al., 1994). The potential of video and computer games is also recognized by parents and teachers who claim that children develop problem solving strategies, learn to recognize complex patterns, and use decision-making criteria in playing games -- reason enough to pay attention to the electronic game phenomenon. But perhaps the most important reason of all to pay attention to electronic games is that it is indisputable that many children find the game culture compelling. Leggo (in press) makes a convincing case for the need to be responsive to the popular culture within the school environment, arguing that teachers can help children make responsible choices as they interact and respond to various popular forms of media.

It is because of the seemingly magical appeal of electronic games, and video games in particular, that I am interested in understanding their allure and capitalizing on their appeal in order to help children learn about math and science. We need to explore the attitudes of parents, teachers, children, and society members in general to see whether both positive and negative beliefs are well-founded, and act in accordance as we take on the enterprise of developing new games. I am convinced there are ways to design electronic games that appeal to children without involving violence, or portraying females in subservient roles. By ignoring the popular culture of video games instead of meeting it head on, we would not only blind ourselves to a strong cultural influence, but lose an opportunity that could have a strong positive impact on math and science education.

References

Dominick, J. (1984). Videogames, Television Violence, and Aggression in Teenagers. *Journal of Communication, 34* (2), 136-147.

Graybill, D. (1985). Effects of Playing Violent versus Nonviolent Video Games on the Aggressive Ideation of Aggressive and Nonaggressive Children. *Child Study Journal,* 15 (3), 199-205.

Leggo, C. (in press). Increasing Teleliteracy: Responsive and Responsible Television Viewing. *Alberta English.*

Lawry, J., Upitis, R., Klawe, M., Anderson, A., Inkpen, K., Ndunda, M., Hsu, D., Leroux, S., & Sedighian, K. (1994). *Exploring Common Conceptions About Boys and Electronic Games.* Department of Computer Science, University of British Columbia, Technical Report 94-1.

Provenzo, E.F. (1991). *Video Kids: Making Sense of Nintendo.* Cambridge, MA: Harvard University Press.

Silvern, S.B., & Williamson, P.A. (1987). The effects of videogame play on young children's aggression, fantasy, and prosocial behavior. *Journal of Applied Developmental Psychology, 8,* 453-462.

PANEL

Multimedia publishing in education: New platforms, products, and markets

MULTIMEDIA PUBLISHING IN EDUCATION: NEW PLATFORMS, PRODUCTS, AND MARKETS

K. B. LEVITAN, Chair
Tidewater Consultants, Inc., 2700 S. Quincy St., Suite 400, Arlington, VA 22206

PANEL DESCRIPTION: With the continued improvements in multimedia hardware and software tools, and the integration of technologies for voice, data, full motion video and graphics, the publishing arena for educational products has been split open and new forces are shaping the marketplace. In addition to traditional textbook publishers, multimedia and the converging computer/communications technologies have brought hardware and software companies, telephone and cable companies, movie studios, and others into the traditional marketplace of textbook publishers. What do these changes mean for education? What will new products look like - for the hand, desk top, client-server network, distance education? How will these new media products be delivered? What kinds of relationships will publishers/disseminators have with educational institutions and their faculty/teachers? How will the purchasing process change? What impact will this have on teaching and learning? Educators need to know the answers to these questions.

This panel brings together a diverse group to explore the changes in multimedia publishing for education from points of view of both traditional textbook publishers and nontraditional organizations entering the publishing arena. Representatives from major publishing and software companies to address the above questions from their own marketing and product development perspectives.

PANELISTS AND PRESENTATIONS:

Scott C. Hardy, Director Media and Multimedia Development, *Harcourt Brace College Publishing, 3700 Commerce St, Suite 3700, Ft. Worth, TX 76102.* **Old Line Publishers in a New World Order: The Role of Traditional Textbook Publishers in Multimedia Instruction**

Felicia M. Woytak, President, *Beyond Books, 5337 College Ave., Suite 453, Oakland, CA 94618*
New Media Brings New Ways of Teaching, Selling and Training

Fred Benz, Manager of Product Marketing, *Kaleida Labs, Inc., 1945 Charleston Rd, Mountain View, CA 94043*
ScriptX: A Cross-Platform Language for Developing and Publishing in Education

Amy Satran, Multimedia Product Development Manager, *Wadsworth Publishing Company, Ten Davis Drive, Belmont, CA 94002.* **New Media and New Business Models in Higher Education**

PANEL

Distance learning

Learning Without Boundaries
An International Overview of Distance/Distributed Learning

ROBERT J. SEIDEL
U.S. Army Research Institute
Advanced Training Methods Research Unit
(seidel@alexandria-emh2.army.mil)

The re-structuring of the U.S. military forces in size and composition, along with decreased funds for real exercises and equipment, has focused attention on how to maintain a well-trained and flexible force. These changes are affecting NATO as well. While NATO for years has been more interested in hardware and weapons technology, recent changes in economics, force structure, and the European infrastructure have stimulated a strong interest in education and training technologies. The Research Study Group on Advanced Technologies Applied to Training Design (RSG-16) was established by Panel 8 of the NATO Defense Research Group, in order to provide an active exchange of data, methods, models, and media throughout the NATO alliance.

While Distance Learning has been developed and used by the open university system for some time, the application to training in the vocational and school-to-work areas have had only a recent focus by Europe. The RSG has been focused on advanced technologies for team, multi-national, multi-service training. Such a focus allows the forging of conceptual links between the traditional approaches of Distance Learning (one-way knowledge dissemination from instructor to learner) and the requirements for distributed training (multi-way dissemination among teams, crews, etc). An RSG-16, NATO workshop was held in Munich, on September 29 through October 1, 1993 and dealt with these and the following topics.

1) Distance Learning/Distributed Training lessons learned - commercial and military.

2) C/E tradeoffs.

3) On-demand instructional value.

4) Impact of instructional culture.

5) Instructional Development requirement.

It was generally agreed at the workshop that Distance Learning has been used for generations. It follows from these topical concerns that the opportunity and the challenge now is for nations to integrate computers and other advanced technologies (e.g., networks, animation, DVI, etc). A modern comprehensive Distance/Distributed Learning program would take advantage of these. A model or approach for representing the various components which must be integrated when designing Learning Systems, was developed at the workshop and provided the framework for my presentation.

Effectiveness and Cost of Distributed Instruction Technology

J. D. Fletcher
Institute for Defense Analyses
Alexandria, Virginia, USA
Fletcher@Charm.ISI.Edu

Interactive courseware (ICW) and video teletraining (VTT) are being used to distribute instruction to physically dispersed learners. This paper briefly reviews what we have learned about their effectiveness and cost.

Effectiveness. Meta-analysis has been used to combine the quantitative results from assessments of ICW. Meta-analysis uses a measure called effect size, which is defined as the difference between the means of two treatment groups divided by either a control group or a pooled standard deviation. Roughly, an effect size of 0.50 suggests an improvement in student performance from the 50th to the 69th percentile.

Two ICW media are considered: computer-based instruction (CBI) and interactive videodisc instruction (IVI). Effect sizes for CBI used in elementary schools, secondary schools, colleges, and adult education have been found to be 0.47, 0.40, 0.26, and 0.42 respectively. Effect sizes for IVI used in colleges, industrial training, and military training have been found to be 0.69, 0.51, and 0.39, respectively. Six studies of IVI compared different levels of interactivity. Effect sizes were higher for the more interactive approach in all six studies.

There are not sufficient data for meta-analyses of VTT. Four assessments of VTT have been performed by the US military using end-of-course achievement as the criterion. An Army study found that 1V/2A and 2V/2A were equally effective and both more effective than residential delivery in providing a specified course. An Army study of its SEP (1V/2A) system found that satellite, residential, and local (using a traveling instruction team) delivery of instruction were equally effective across 11 courses. A Navy study of a 2V/2A system found residential and remote delivery of 13 convenings of a single course to be equally effective. An Army study of a 1V/2A system found remote to be more effective than residential delivery on 3 of 4 outcome measures.

Costs. Since 1979 at least three reviews have concluded that the cost of CBI is less than the cost of conventional instruction due to an average 30% savings in student time. Cost ratios for 13 IVI programs were found. All 13 were lower than 1.00 indicating lower costs for IVI. Five ratios were for initial investment costs and averaged 0.43; eight were for operating and support costs and averaged 0.16. A times savings of about 30% was also found for IVI.

The Army SEP study of 1V/2A found costs per student week for satellite, residential, and local delivery to be $874, $1182, and $376, respectively. The Navy study of 2V/2A reported savings of $68,721 for the 13 remote convenings of their course. An Army study of costs alone to deliver a course found analog satellite, digital satellite, local, and residential delivery to be $611, $419, $877, and $2012 per student, respectively.

The issue should not be just _if_ ICW and/or VTT should be used but _how_ they can best be used to obtain the instructional results needed. The results thus far are promising, but conclusive evidence awaits more studies in which both cost and effectiveness data are collected under the same evaluation design.

An Air Force Perspective on Distance Learning

J. Michael Spector, Ph.D.
Armstrong Laboratory
Instructional Design Branch Senior Scientist
(spector@alhrt.brooks.af.mil)

The Air Force Air Education and Training Command (AETC) conducted an extensive baseline study of distance learning technologies in academia, industry, and government in 1991. Basic findings indicated that many organizations were using various forms of distance learning ranging from satellite delivery of course materials to distribute stand-alone computer-based lesson materials. The basic motivation for using distance delivery technologies was to save travel costs associated with training. Little innovation existed in the selection and sequencing of distance learning training materials.

Armstrong Laboratory has hosted two distance learning symposia to maintain currency with state-of-the-art practice in distance learning. The most recent of these symposia was held in Denver in October 1993. The trend reported in the earlier 1992 symposium was to use satellite technologies to replicate the classroom at a distance with as much fidelity and as little expense as possible. The 1993 meeting, however, revealed strong interest in using distributed networks as an integral part of instructional delivery. Satellite delivery is based on a mass communications model and it certainly has appropriate uses, both in terms of cost effectiveness as well as in distance delivery, however, is based on an interpersonal communications model. The difference in communications models implies a likely difference in areas of optimal utility. Where one kind of technology might be weak the other may be strong.

The differences in mass and interpersonal communications models suggests that consideration be given to rigorous testing of a wide variety of distance learning strategies, ranging from collaborative networked environments to very high fidelity two-way audio/video transmissions, across a variety of student profiles and subject matter. The point of conducting this research is to establish a guide for effective integration of technologies into instructional development. Current Air Force R&D in this area is explicitly directed to this last concern.

Implementation Issues in Distance Education

GREG KEARSLEY
Dept. Educational Leadership, George Washington University
Washington, DC 20052 (kearsley@gwuvm.gwu.edu)

Distance education is hardly a new development; correspondence study began in the 19th century. Instructional television, which is now experiencing a resurgence in the form of satellite delivery, is an innovation of the 1960s and computer-mediated communication (aka networking) appeared in the 1970s. A number of large-scale distance learning institutions and programs have been in operation for a decade or two. So we have had many years to learn how to do distance education successfully.

What have we learned? First, we have learned that distance education is a complex undertaking and requires a major overhaul of the existing infrastructure designed for classroom teaching. Teachers need to teach differently (and hence need to be trained with new skills). A high degree of planning and preparation of curriculum and materials is needed in any form of distance education, unlike the "seat of the pants" mode of most classroom teaching. Distance learning programs need different administrative procedures and structures than those currently used in traditional schools. For example, learner support (especially tutoring and counseling) is a critical component of successful distance education -- but not one that receives much attention in conventional classroom instruction. Finally, distance education tends to be capital intensive, requiring considerable initial funds for program development and acquisition of delivery systems. This usually means cost-benefit studies and business plans -- analytical methods that few educational administrators are comfortable with.

The successful implementation of distance education requires a systems approach -- careful attention to all instructional and administrative components and their interactions. Even small-scale efforts at distance education require major changes in the way things are done. Being aware of all these changes, and figuring out how to make them, is the biggest challenge of distance education.

Short Papers

Use of Multimedia Development Software for Engineering Courseware

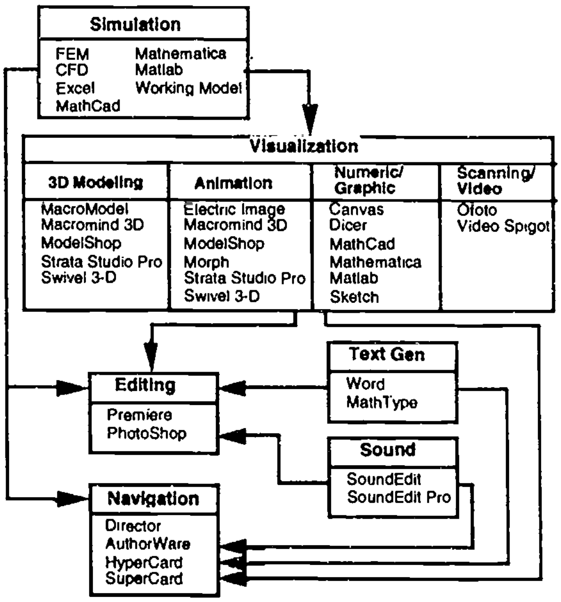
R. ABBANAT , K. GRAMOLL, J. CRAIG
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150

Over the last several years, integration of multiple media sources into a single information system has been rapidly developing. With the recent introduction of many high level authoring, animation, modeling, and rendering programs for personal computers, significant multimedia courseware programs can now be developed by professors and students. Even with these new tools, a considerable amount of time is required to produce an interactive multimedia program for engineering education. This paper examines various multimedia development tools and how they can be used for engineering education courseware.

With support provided by the NSF SUCCEED Coalition, we have initiated a research and development effort to produce engineering courseware using various multimedia technologies. The primary goal of the courseware is to illustrate a number of basic but highly abstract concepts in introductory engineering mechanics courses using integrated video, speech, and three-dimensional graphics in self-contained software modules. These modules supplement lectures and textbooks, and illustrate topics not readily accommodated in the printed format of a book. For example, data visualization can be an important factor in multimedia courseware to enhance concepts describable only in words or symbols in a textbook or lecture. The paper discusses our experience with currently available authoring, modeling, rendering, animation and data visualization software.

The engineering concepts best suited as a topic for an engineering courseware module are those which are highly abstract and difficult to visualize, such as a 3D stress tensor, or are dynamic, such as complex structural dynamic response. Concepts which are static (a two dimensional bending moment diagram) can still be presented but often are not as stimulating. On the other hand while a simple 2-D bending moment diagram can very easily be shown in a textbook, multimedia methods can link it to a video of a practical situation or illustrate how changes in loading affect the diagram. The presentation of highly abstract concepts which may have no physical form can be facilitated and enhanced using multimedia. Consider, for example, using multimedia to illustrate the behavior of a 2^{nd} order tensor as it is used to represent stress or strain. In this case the multimedia techniques used may be closely linked to scientific visualization methods.

The development process for a courseware module is highly nonlinear (see figure), and although numerous multimedia software programs are needed, they can be separated into categories based on functions performed, e.g., simulation, modeling, rendering and authoring. Once an engineering topic has been chosen for implementation, the developer will in many cases, need to first simulate a problem or situation that is the engineering focus, then transform output from the simulation into an appropriate visual form, define animation to describe the system, render the animation, generate audio and text components, define the navigation topology and finally integrate everything using an authoring package. Depending on the nature of the topic, not all of the processes listed above are required for every module. Frequently a single program has the capability to perform many of the tasks simultaneously.



Our presentation describes experience with development of more than 12 modules. Several of these modules are demonstrated to illustrate some of the development issues.

Development of a Multimedia Interface Compatible with Changing South African Student Demographics

ALAN AMORY, AND LYNLEY MACKENZIE
Department of Biology
University of Natal, Durban, 4001, South Africa

DEREK WATT
Department of Botany
University of Durban-Westville, Private Bag X54001, Durban, 4000, South Africa

Inequalities inherent in the South African secondary education systems, directly attributed to apartheid (segregated schooling for white, black and indian South Africans), have rendered meaningless traditional University admission requirements. As a result, many tertiary education institutions have adopted progressive affirmative action policies to increase the accessibility of higher education to previously disenfranchised communities. The success of such policies requires the development of innovative and dynamic teaching methods that would cater for the academic diversity of both present and future student populations. Such mechanisms must allow for self pacing, while not marginalising into remedial programmes those students with the potential to succeed. Such was the promise of traditional computer based training (CBT). To date, the impact of CBT modules in South African tertiary education has been minimal. This may be due to the lack of vividness or fidelity of the presentations and may be overcome by a multimedia approach. Hence, the aims of this research were to: 1. Involve the target group (first-year science students) in the development of multimedia programmes; 2. Investigate possible cultural biases within multimedia presentations; 3. Attempt to identify the visual literacy and computer skills of these students; and 4. Discern the most useful and easy-to-understand icons required for clear communication with the target group. Questionnaires were used to determine the ability of biology students, from both historically 'black' and 'white' Universities, to identify and design icons (paper-based, 349 respondents). In addition, colour and voice preferences of these students were identified (computer-based, 65 respondents).

Indian students exhibited a greater ability to interpret 16 icons than did white or black students. Similarly, they outperformed the other two groups at pictorial representation of command or options. This capacity of indian students appeared to be correlated to a number of factors including greater access to and training in computers, more frequent viewing of television and movies and only sporadic reading of books, when compared to the other two groups. Furthermore, all students had a greater capacity to identify icons than to devise them. Some black student outperformed, or were equally competent to, white students in this exercise, although the percentage of non-response by black students was high. There appeared to be little intuitive understanding of iconic information among the groups tested, in that students preferred icons containing words and, therefore, reinforcing and enhancing meaning. Computer literacy was not developed highly. We identified icons that appeared to be understood by most students, which included: Next Page, Previous Page, Video and/or Movie, Start, Stop and Fast Forward. In addition, we obtained a number of icons better suited to the representation of Picture and Listen. Icons for Help, Dictionary and Index will have to be taught, while it will be necessary to develop new representations for Go to, Go Home, Exit, Menu and Memo. A sub-sample of students was used to identify voice and colour preferences. Most student selected a clear female voice without a discernible accent. Blue was the most popular colour, followed by red.

We believe that any user interface developed for use by our students will require not only pictorial representation of commands or options, but should contain text that clearly identifies the choices available. Also, it will be necessary to teach students how to use such software, especially as the syntax of operating the mouse-button is closely allied to the use of an iconic interface.

Harmony: A Tool for Navigating Through Deep Hyperspace

KEITH ANDREWS, FRANK KAPPE, AND JÜRGEN SCHIPFLINGER
Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology, A-8010 Graz, Austria

Harmony is a new tool conceived for navigation through *deep hyperspace*, i.e. large-scale (many tens of thousands of nodes), dynamic hyperstructures. One of the main problems of such hyperstructures is disorientation - the "lost in hyperspace" syndrome. Harmony draws on the real-world experience of "finding your way in an unfamiliar part of the world" to suggest promising approaches to the task of navigating through deep hyperspace.

The underlying infrastructure of a *Hyper-G* (Kappe, 1993) server provides the basis for Harmony's navigational metaphors. Based on the real-world observation that a hierarchical stack of progressively more detailed maps (e.g. The World - Canada - British Columbia - Vancouver) is better than a single global map, Hyper-G provides for the hierarchical structuring of hypermedia documents into collections and subcollections in addition to (and orthogonal to) the traditional plane of link-based browsing. Harmony's navigational facilities include:

- **Hierarchical Navigation:** Harmony's *collection browser* displays the tree structure of the collection hierarchy, opening up new levels of detail as the user navigates down the hierarchy. Subcollections can be opened and closed and documents viewed by double-clicking. The collection hierarchy's representation is automatically expanded to show the location of documents reached by other navigational techniques (search, local browsing, etc.). Such *location feedback* is an important feature of Harmony not found in other comparable systems, allowing users to build up knowledge of the location of documents within hyperspace.
- **Search:** Harmony supports both attribute (title, keyword, etc.) and full text search, performed by default in the current collection. A ranked list of matching documents is displayed, from which individual documents may be selected. This is similar to looking up a historic building in the index on the back of a city map.
- **Local Browsing:** Local browsing refers to the process of following hyperlinks from within a document. Harmony presently supports linking from text, image, and 3D scene documents. Local browsing can be compared to visiting a historic building and then wandering down a few nearby streets.
- **Local Map:** A "local map" can be generated to show documents related to a particular document by hyperlinks. By default, two levels of incoming and outgoing links are presented. The local map is active: users can double-click on documents to view them. In our analogy, a local map corresponds to a kind of short-range radar, showing the vicinity around the current document.
- **Information Landscape:** Harmony's information landscape is a 3D representation of the collection hierarchy. Users can "fly" over the landscape looking for salient features, like taking a helicopter flight over a city and picking out its important buildings.

We believe this combination of new and traditional features helps alleviate the sense of disorientation commonly experienced when navigating deep hyperspace.

References

Andrews, K., & Kappe, F. (1994). Soaring through hyperspace: a snapshot of Hyper-G and its harmony client. To appear in *Proc. of First Eurographics Symposium on Multimedia, Graz, Austria.*

Kappe, F. (1993). Hyper-G: a distributed hypermedia system. In Leiner, B., editor, *Proc. INET '93, San Francisco, California* (pp. DCC-1-DCC-9), Internet Society.

A New Language for Defining Sample Answers in Authoring Systems

CHRISTIAN BACHER AND THOMAS OTTMANN
Institut für Informatik
Universität Freiburg, Rheinstraße 10–12, D-79104 Freiburg, Germany

Authoring systems for courseware development usually contain answer judging modules as constituent parts. Entering a text string as an answer to a posed question is a standard option. This so called free form text question/answer facility is usually based on purely syntactic principles. Thus, the author specifies a class of legal sample answers by using an appropriate specification language. The system classifies student answers by syntactic comparisons with the sample answers.

In current authoring systems (like Authorware Professional[2], Autool[3]) an author typically may require that certain (key-)words must appear in a legal answer, he may define synonymes and make use of several options, like ignoring word order, punctuation, capitalization, blanks etc. Difficulties, however, occur when the author wants to characterize words or phrases as forbidden for a correct answer. A further desirable property of the answer judging module is to tolerate typing errors in student answers up to a reasonable extent.

We describe a new specification language, called TAL (TRAIN authoring language) which provably exceeds the expressive power of similar languages. It in particular allows to forbid the occurrence of words in parts of legal answers. TAL is defined inductively like a logical language. Thus, using TAL an author may require that " 'red' and 'yellow' or 'blue' but not 'green' should occur in the answer".First, the author can require that specific strings of phrases (defined by means of patterns) must occur somewhere in or at the beginning of an answer. For this purpose the elementary expressions of TAL are used. If φ and ψ are TAL expressions, then $[\varphi \& \psi]$ and $!\varphi$ are also TAL expressions. An answer matches with $[\varphi \& \psi]$, if it matches with φ and with ψ; an answer matches with $!\varphi$ if it does not match with φ. Furthermore, TAL contains with expressions φ and ψ also the order expression $0(\varphi, \psi)$; an answer X matches with $0(\varphi, \psi)$, if the beginning of X matches with φ and the rest of X starting at the position where φ ends matches with ψ. A more precise definition requires to define for every answer X and TAL expression φ precisely an end-position "end of φ in X". This definition has also to take into account that typing errors up to a limit specified by an author should be tolerated in student answers. The following events in a text string are considered as typing errors: missing of a character, a superfluous character, a wrong character, or the interchange of two adjacent characters.

We illustrate the use of TAL by a number of examples. Then we show that TAL can simulate the specification languages of the above mentioned authoring systems. Finally, we report about the implementation of TAL as part of a new prototype authoring system TRAIN[1]. The evaluation of expressions $\varphi \in$ TAL requires the implementation of a recursive evaluation function which takes φ and a student answer X as input and produces the output 'true' if X matches φ. For the evaluation of the elementary TAL espressions Ukkonen's improved dynamic programming algorithm for approximate string matching[4] and a version of the shift-or-algorithm for pattern-defined phrases[5] has been implemented.

The new specification language allows to specify very complex sets of answers even when using the logical operators alone.

References

[1]**F. Augenstein, Th. Ottmann, and J. Schöning.** Logical Markups For Hypermedia Dokuments. *Proceedings ED-MEDIA '93*, 17-25.

[2]*Authorware Professional Reference Manual.* Minneapolis MN, 1990.

[3]**J. Garratt and H. Maurer.** *Autool Version 2 Manual for COSTOC Authors.* IIG Graz, 1987.

[4]**E. Ukkonen.** Algorithms for approximate string matching. *Information and Control*, 64:100–118, 1985.

[5]**S. Wu and U. Manber.** Fast text searching allowing errors. *CACM*, 35(10):83–91, 1992.

Dynamic Media for Electronic Books

Philip Barker
Interactive Systems Research Group
Human-Computer Interaction Laboratory
University of Teesside, Cleveland, UK

Electronic books have become an important mechanism for the storage and delivery of multimedia information within a variety of different contexts. In situations where their use is appropriate, there are obviously many advantages to using electronic books instead of conventional ones. Some of the more important reasons for the growing use of 'electronic publication' is the substantial volume reductions that can be achieved (compared to publication on paper), the ease with which electronic information can be accessed and shared, the flexible ways in which it can be delivered and the ease with which it can be transferred from one geographical location to another. Of course, electronic information has two other important assets - it can embed interactivity and it can be highly dynamic (that is, it is able to change its form and content to fit particular needs and requirements).

The organisation of information within electronic books can be based upon the use of both linear and hypermedia structures. The particular approach that is used will usually depend upon the purposes for which a given electronic book is to be used. Although many electronic books have been published in a linear format, increasingly, there is now a move towards the more extensive use of hypermedia structures - primarily, because of the added flexibility that this approach can be used to achieve.

The types of information contained within a given electronic book will be dictated by the needs and requirements of the individual application for which it is to be used. Sometimes, the use of pure text will be sufficient. However, augmenting text with diagrams and static pictures is also a fairly easy thing to accomplish. Of course, the advent of multimedia personal computers and consumer equipment for handling sound (such as the Sony Data Discman and Philips' CD-I) means that sound is now becoming a valuable resource in many electronic books. Many publications are now also able to embed moving pictures - for example, the McGraw-Hill Encyclopedia of Mammalian Biology.

There are two basic approaches to the provision of moving picture sequences within electronic books: animation and digital motion video. Each approach has its associated attractions and limitations. Each also has a context in which its use is most appropriate. Indeed, there is a complex ensemble of interacting factors that must be considered when attempting to decide whether to use an animation sequence or a video sequence for a particular purpose within any given electronic book publication. We have been studying the use of these resources in different situations in order to explore how the important factors interact and influence each other. The intent of the work has been to formulate a prescriptive model which can be used to recommend which type of resource to use in any given situation. Some of the types of application that have been studied include: interactive training manuals; electronic brochures; on-line technical manuals; interactive video tutorials and simple surrogations based on virtual reality.

References

Barker, P.G., (1991). Electronic Books, Special Edition of *Educational and Training Technology International*, 28(4), 269-368.

Barker, P.G., (1992). Electronic Books and Libraries of the Future, *The Electronic Library*, 10(3), 139-149.

Barker, P.G., (1993). *Exploring Hypermedia*, Kogan Page, London, UK.

Barker, P.G. and Collis, B., (1993). Changing Perspectives in Book Technology: Electronic Books in the Adult Education and Training Setting, *Computers in Adult Education and Training*, 3(2), 5-20.

Catenazzi, N., (1993). *A Study into Electronic Book Design and Production: Hyper-Book and the Hyper-Book Builder*, PhD Thesis, Department of Information Science, Strathclyde Business School, University of Strathclyde, Glasgow, Scotland, UK.

Richards, S., (1993). *End-User Interfaces to Electronic Books*, PhD Thesis, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Distanciation versus Illusion: a key issue in educational multimedia

C. BEARDON AND S. WORDEN
Rediffusion Simulation Research Centre, Faculty of Art, Design & Humanities, University of Brighton, Grand Parade, Brighton BN2 2JY, England

Educational multimedia systems should not only provide access to off-the-shelf information, but should develop the critical awareness of students and their ability to transform what is given into something new. The concept of the Virtual Museum is an attempt to electronically recreate both the institution of the museum and its contents. It aims to reproduce the aura of the object, the prestige of the public gallery and the authority of the expert. In the Virtual Curator (Beardon & Worden, 1993) the user is a curator with a store of unexplored objects. Objects can be copied, assembled into groups (e.g. a poster, or a display cabinet) and arranged within a virtual exhibition space. Any such assembly can also be put back into the store for future re-use and new objects can be added. The user's task is to mount an exhibition on a theme. Using the Virtual Curator a student appropriates cultural images in order to construct her own point of view from her own perspective.

Many multimedia systems imply an illusory (virtual) reality inside the machine engendered by the use of photorealistic images, but photorealism in turn implies a single viewpoint from which the whole scene is authoritatively viewed. The significance of all objects in the scene, involving their position, their orientation, their relative size and the extent to which they are hidden, is determined by the location of this viewpoint. Objects are not displayed according to an understanding of their construction or significance but with respect to the single all-knowing eye, something that has been central to European culture since the Enlightenment.

Brenda Laurel compares the computer's illusory world to the theatre, with agents who act and directors who control the action (Laurel, 1991). She does not adopt a critical approach to multimedia but encourages the audience to willingly suspend belief in order to be passively healed and strengthened. Bertholt Brecht was opposed to the concept of a theatre of illusion - one which encourages users, "to read the proffered text naively, as a mirror-image of a preexisting world" (Wright, 1989, p.56). Brecht developed the concept of "distanciation" to refer to the creation of a critical distance between audience and play in order to break the illusion and make the audience aware of what is real. "Distanciation is not a style or aesthetic gambit but an erosion of the dominant structures of cultural consumption ... " (Pollock, 1988, p.163).

Brecht developed several techniques to achieve distanciation: events are represented so that the audience feels instant recognition; the play is fragmented into a number of discrete episodes that are interrupted and do not form a continuous unbroken narrative; the stage design is not naturalistic; collage and montage are used. These are good guidelines for the development of sophisticated educational multimedia software. They have been embodied in the multimedia system "Queen's Park" where the objective is both to explain the history of a local park and to explain the process of creating history. A guide is used who tells the story of constructing the history and various distanciation techniques are employed. For example, photographs are presented in a way that encourages a double reading as both part of the process and part of the product. Also, there are times when text and image interact to raise questions about the nature of historical enquiry.

References

Beardon,C & Worden,S(1993) Computers and the visualisation of culture. *Computer Science, communications and society: a technical and cultural challenge* Conf. Sep. 22-24 1993. University of Neuchâtel, Switzerland.
Laurel, B. (1991) *Computers as theatre*. Addison-Wesley, Reading, Mass.
Pollock, G. (1988) *Vision and Difference: Femininity, feminism and the Histories of art*. Routledge, London.
Wright, E. (1989) *Postmodern Brecht: a re-presentation*. Routledge, London.

Integrated Telephony in Executive Education - the Auckland MBA Experience

KATHY BOARDMAN
Management Science & Information Systems, University of Auckland, Pvt Bag, Auckland, New Zealand
e-mail : K.Boardman@Auckland.ac.nz

PETER M. STEVENS
Apple Computer New Zealand, Private Bag, Ponsonby. Auckland, New Zealand.
e-mail : NZSTEVENS@AppleLink.Apple.Com

Competition in the executive education market in New Zealand provided the motivation for the University of Auckland to improve the quality and efficiency of the Executive Master of Business Administration (EMBA) learning process in 1993, via a project involving the integration of telephone technology and Apple PowerBook computers.

The objective of the EMBA programme is to impart knowledge, skills and values to relatively computer illiterate students through a learning process which comprises large group, small group and individual activities. For the EMBA students the combination of one day per week on-campus instruction in a lecture room environment, off-campus preparation for group and individual assignments and work and family responsibilities results in inevitable scheduling difficulties.

A proposed solution was to enable programme participants to work anywhere, anytime, both individually and with their colleagues, and to be effectively "wired up" to the university. Scheduling difficulties associated with the group learning process would be mitigated by enabling students to use the telephone system to transfer files, send/receive mail, access databases, engage in real-time conferencing and use telephony-aware groupware.

Various hardware and software suppliers were asked to submit proposals. The Apple Macintosh PowerBook was selected because its intuitive user interface was ideal for users with little computer experience, and AppleTalk Remote Access (ARA) software allowed users to access a network and transfer text, graphic, audio and video files over standard telephone lines - all at the click of a button. Students were required to purchase a PowerBook, 14,400bps data/fax modem, printer, ARA, QuickMail, Microsoft Word and Excel and Aldus Persuasion. EMBA faculty were similarly equipped.

Initially AppleLink provided a bulletin board service which enabled participants to exchange files, communicate news items and send electronic mail using a very simple interface. However, since the objective was to provide more than just a bulletin board ARA was next used to enable direct communication between remote Macintoshes and campus network services. Dial-in access to the university network was achieved using a nine-line hunt group on the University's Ericsson telephone exchange which connected to three Cayman Systems GatorLink ARA servers. These servers were directly connected to the ethernet backbone and a Macintosh Quadra functioned as a mail server and data repository for general access to information and secure areas for storage of individual and syndicate group files. Farallon's Timbuktu enabled remote server administration and Claris FileMaker Pro was used for help desk information and call logging. NCSA Telnet provided access to the university's IBM ES9000 mainframe-based "Notis" library catalogue. CE Software's QuickMail allowed electronic mail exchange among programme participants and other university staff members and the QuickConference feature enabled groups to engage in real-time exchange of information.

The next phase of the project will involve the introduction of groupware, such as Group Technologies' Aspects and ON Technology's Instant Update, which will make it easier for students to collaborate on documents in a coordinated manner. It is in this area of electronic collaboration that we will be able to truly exploit the potential of distance education. Although e-mail facilitates the preparation of group projects edition control can become a nightmare, and even the "cut" and "paste" of modern software can become cumbersome when multiple authors are working between multiple versions of the same document.

A key question is whether the integration of telephone technology with personal computers has improved communication and learning. The short answer is yes. The computer has become an indispensable tool and many students applied the technology to their business environment. Staff and student presentations have become more interactive and professional. Collaboration has been made easier and more exciting. In terms of the success of the EMBA programme applications for places in the 1994 programme have increased and in contrast with past trends the majority of applications are for the EMBA programme.

Intelligent advisor systems and visual cues

MARIE-MICHELE BOULET
Dept. of Computer Science
Université Laval, Québec, CANADA G1K 7P4

Among the mandatory courses of the undergraduate curriculum in Computer Science at Université Laval, there is *Systems analysis and design*. Information modeling, and more specifically Entity-Relationship approach (Chen, 1976), is an important topic covered in that course. Data modeling is not easy. The learner's task in data modeling is to figure out what types of data underlie the business function under study. The learner must also capture a replica of that structure. Each business function is different. Leshin, Pollock, and Reigeluth (1992) stated that: "Transfer tasks cannot easily be broken down into steps, because the activity varies each time the task is performed" (p.82). Elaborating an ERD corresponds to a transfer task. The *Systems analysis and design* course has two parts. Concepts, principles, and rules are taught during a theoretical part. Learners transfer this knowledge during a practical part; they have to resolve cases. The COnceptual DAtabase Modeling Advisor (CODAMA) was developed to further the transfer of knowledge during the practical part.

Summary of visual elements

Students enrolled in *Systems design and analysis* course have weak knowledge of the domain and no practical experience. They draw their ERD with a CASE tool. CODAMA is interfaced to that CASE tool. The CODAMA's visual elements are the following (Boulet, 1992):

ERD or part of an ERD is used. Each ERD refers to several kinds of documents used within the organization (such as forms, memos, lists).

Underlined words allow the recall of prerequisites in the short term memory. Learners use the mouse to select the prerequisites to be explained.

A network of prerequisites is presented in different ways. A learner can ask to visualize the explanations related to a particular prerequisite. In that case, the related part of the network is displayed. Another learner may want to know where she or he is within the network. CODAMA will take into account several data collected in regard of explanations requested and mistakes, to make an hypothesis about the level of knowledge. The corresponding part of the network will be displayed.

A hierarchy of prerequisites can be displayed. A learner can ask to which objective a particular question or mistake is related. CODAMA will show where she or he is within the hierarchy. Another learner can ask which objective(s) is (are) not mastered. CODAMA will display the hierarchy. CODAMA can also provide an answer to a learner asking for the list of prerequisites to a concept or a principle. It can also show where the learner is within the hierarchy; it will make an hypothesis based on several recorded data.

Conclusion

Students, when asked what is an entity or a relationship, use less textbook like explanations and more examples. So, the advisor favors the transfer of knowledge. At the end of their curriculum, students have a training period within an organization. More than sixty percent asked to use CODAMA during this training period. A part of this success is due to the various uses of images and visual cues. They have to be carefully thought and planned during the development life cycle.

References

Boulet, M.M. (1992). *Advisor systems: The development life cycle illustrated by two case studies.* Québec: Edition Dwayne.

Chen, P.P.S. (1976). The entity-relationship model - toward an integrated view of data. *ACM Transactions on Database Systems.* 1, 9-36.

Leshin, C.B., Pollock, J., & Reigeluth, C.M. (1992). *Instructional design strategies and tactics.* Englewood Cliffs: Educational Technology Pub.

A Computer-Based Learning Environment For Investigating Skills, Learning, and Teaching In Technical Listening

A. BREULEUX
Faculty of Education
McGill University, 3700 McTavish, Montréal QC, H3A 1Y2, Canada
e-mail: ed13@musica.mcgill.ca

R. QUESNEL
Faculty of Music
McGill University, 555 Sherbrooke St. W., Montréal, QC, H3A 1E3, Canada
e-mail: quesnel@music.mcgill.ca

This paper presents the development of a computer-based training (CBT) environment (Timbral Ear Trainer II) for timbral ear training, a specialized form of ear training aimed at sound engineers. The purpose of developing the environment is threefold: improve learning and teaching in timbral ear training, improve our understanding of the perceptual and cognitive skills associated with the domain, and improve our understanding of principles and fundamental mechanisms involved in learning and teaching. We present a set of initial assumptions about the skills involved, the learning mechanisms, and the training strategies that are appropriate for this domain. We then present a number of ways to implement versions of these assumptions and to test the relative effectiveness of these alternatives.

We are developing Timbral EarTrainer II in part to gain a better understanding of principles of learning and teaching, both within the specific area of timbral ear training and more generally. We describe our environment by systematically articulating choices that we make among theoretical, instructional, and technological alternatives. We also propose to test some of these design alternatives to investigate aspects of the phenomena that the environment is supporting, i.e., learning a specialized skill.

One of the main responsibilities of a sound engineer is to control the quality of the sound during the recording process. An important qualitative aspect of a recording, besides the musicality of the performance, is its timbral content. Proper level and spectral balance between the mixed elements of the recording and the absence of extraneous noises and distortion are key elements. The sound engineer needs also to associate timbre differences and desired sound qualities to objective parameters of sound commonly under his or her control. Therefore, timbre perception acuity and memory as well as listening strategies are essential skills for sound engineers. Analyses of these skills form the basis for assumptions that can be articulated and implemented in Timbral EarTrainer II.

The environment being developed is hybrid, using a variety of teaching methods depending on the nature of the skill being trained. For example, the memorization of an initial set of nine timbre categories requires extensive practice. For this part of the training, drill and practice exercises are used. In contrast, the development of effective listening strategies requires guided practice. Therefore, this part of the training is based on an apprenticeship model using coaching.

A research module is being developed as part of the learning environment to test our various assumptions. For example, different instructional sequences will be tested against each other. Different interaction modes, giving the students more or less freedom in navigating through the program, will be tested. The initial set of assumption stresses practice as the main learning mechanism. We will test how the presentation of theoretical material (e.g., on auditory perception, timbre perception) might contribute to the learning. We will also test the effectiveness of letting students control the difficulty level of the exercises. Different interface designs will also be tested.

The effectiveness of the different designs in improving the students' acuity and memory for timbre and their listening strategies will be verified using a pre-test / post-test procedure. The performance level of the students is evaluated both in terms of accuracy and speed of execution (e.g., long-term memory for timbre categories, the strategies used in complex listening tasks). The data collection methodology is non-intrusive: a "spy" module records all actions performed by students during their work.

An Inteactive Case Study in Strategic Marketing: An Examination of Some Conceptual, Design, Development and Testing Issues

RICHARD BROOKES and SIMON DIXON
Department of Marketing and International Business
The University of Auckland, Auckland, New Zealand

Cases are integral to most marketing courses (Kotler and Armstrong 1991; Cravens and Lamb 1993; and Jain 1993). They help bridge the gap between the formal class-room environment and the highly pressured, practical world of marketing. Cases require students to sift through a mass of material; to identify strengths, weaknesses, opportunities, threats and key issues facing the organisation; to specify and evaluate alternative courses of action; to make recommendations; including the expected costs and benefits of implementation. In the process, students must demonstrate a sound grasp of theoretical and/or conceptual issues under-pinning "real-life" situations. Weaknesses of the traditional "paper-based" case approach are the static nature and presentation of information and that students progresses through the case in the same sequence, rather than according to their cognitive and/or perceptual preferences for assimilating information.

The Royal New Zealand Ballet Company was selected as the Department's first multi-media case, both because of its challenging nature and because of the extremely "visual" material. It provides all the elements of the traditional approach to case study work plus the advantages of an "interactive learning system" in that it is multi-centred; it includes multi-media; and it is in a multi-modal format (Barker 1991). The case structure is relatively non-directional. Sections in the Menu box are neither numbered nor suggestive of priorities (Figure 1). Students progress through the Menu items according to their own priorities, predispositions and/or on-going feed-back results. At any stage they can access additional information, pictures, video and/or audio clips which supplement the material in the primary box, by clicking on any word or item that is in Bold type, to bring up a secondary box (Figure 2). Students need to judge for themselves how much this material is "worth" in terms of the amount of time and effort they devote to it. Finally, students can delve to various levels of analysis, depending on the "costs" they associate with analyzing the material to a level and standard sufficient for the task at hand.

Figure 1.

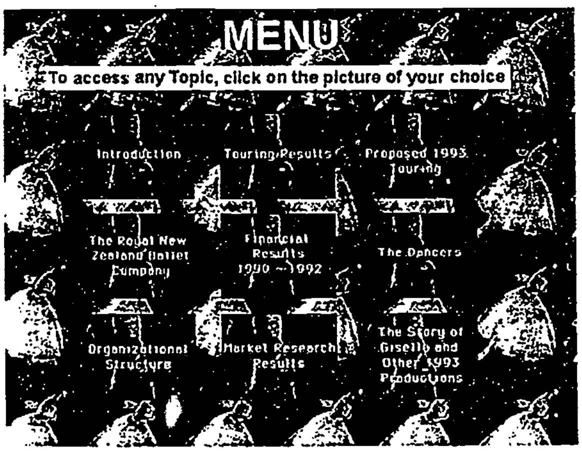
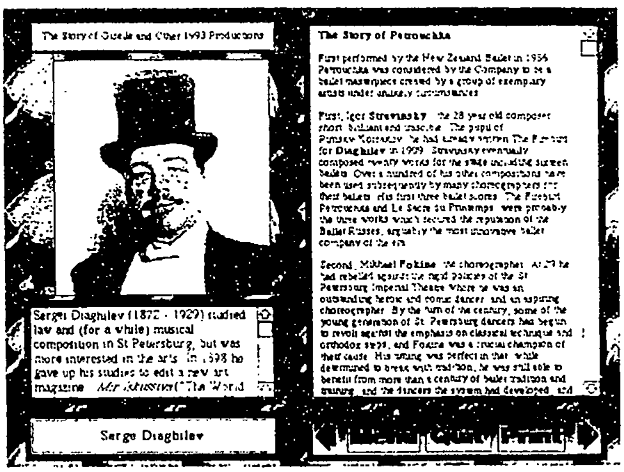


Figure 2.



References:

Baker, P., "Designing Interactive Learning Systems", 1991, ETTI, Vol. 27, No. 2, 125145.
Cravens D. and Lamb, T., Strategic Marketing Management Cases, 1993, Irwin.
Kotler, P. and Armstrong, G., Principles of Marketing, 5th ed., 1991, Prentice-Hall.
Jain, S. C., Marketing Planning and Strategy, Instructor's Manual, 4th. ed., 1993 South-Western.

Adaptive Hypermedia in an Intelligent Learning Environment

P. BRUSILOVSKY
International Centre for Scientific and Technical Information
Kuusinen str. 21b, Moscow 125252, Russia. E-Mail: plb@plb.icsti.su

For a number of years we have been investigating the problem of creating intelligent learning environments (ILE) which integrates an intelligent tutor, a learning environment, and a hypermedia system. Such an integrated system can support learning both procedural and declarative knowledge and provide both guided and student-driven styles of learning. Our position is that an integrated system should be not just a sum but a real integration of its components. In particualr, we propose that the results of students' work with any of the components during the session are to be reflected in the central student model and then taken into account by other components to adapt their performance to a particular student.

This paper presents our experience in creating an adaptive hypermedia component for ILE. There are two ways for a hypermedia component to adapt its work to the student's knowledge and learning goals. First, it can adapt the content of a hypermedia page, providing more explanation for newly learned concepts, shortening it for well learned concepts and sometimes hidding references to concepts which are not ready to be learned. Second, it can adapt the layout and the content of the list of hyperlinks to related nodes, providing an implicit guidance by reordering or marking the hyperlinks. The latter way is also known as *adaptive navigation*. There are several known works in the world devoted to both the ways of adaptation. We propose a special approach for creating an adaptive hypermedia in ILE which supports both the ways of adaptation. The main features of our approach are the following.

- The central part of the hypermedia network is designed as the visualized and externalized domain structure network (concept map).
- All teaching operations are directly represented in the hypermedia network and interlinked with all related concepts.
- An external hypermedia representation of a concept or a teaching operation is not stored, but generated or assembled from its internal frame-based representation.
- The content of the hypermedia page is adapted to the student knowledge reflected in the student model.
- The hyper-links from general index and from any node to related nodes are visually marked reflecting the current "educational state" of the related nodes for the given student.
- Student interaction with the hypermedia component is reflected in the student model and can be used by other components of ILE.

We have designed several ILE for different subjects which employ adaptive hypermedia. These systems use original adaptation techniques. More information about our work on adaptive hypermedia can be found in (Brusilovsky et al, 1993).

References

Brusilovsky,P., Pesin,L., & Zyryanov,M. (1993). Towards an adaptive hypermedia component for an intelligent learning environment. In Bass,L.J., Gornostaev,J. & Unger,C. (eds.) *Human-Computer Interaction*. Berlin: Springer-Verlag, 348-358.

A Windows-Based Speech Aid and Language Learning Tool For the Speech Impaired

DAVID J. CALDER, ROGER W. SMITH, ROBIN A. HOSIE
Curtin University of Technology
Perth, 6001, Western Australia

Computer-based speech aids for the vocally impaired use indirect activation (e.g. iconically labelled keys) to reproduce synthetically or naturally generated vocal sounds. Our current research exploits existing flexible design models for development as language learning tools that may assist the speech therapist. Most vocally handicapped people suffer other forms of handicap; either cognitive, physical or both. System design for such a broad spectrum is therefore complex and it is a folly to assume any vocal handicap in isolation. Speech aids should offer some alternative means of input based on remote switches [1]. Where the user is quadriplegic or possibly suffering from varying degrees of spasticity, there is a need to transfer the functions of all the keys on a standard keyboard into one or two switches. Algorithms using predictor techniques have been investigated in order to improve user models [2].

User interface models

A touch screen system was built to test software linking a speech output to a softkey input. Various multi-level screen structures were explored. It was decided to create a program suite with two main components. One allows easy configuration of the user touch panel so that the human interface can be adapted to the specific needs of the patient. The other acts in accordance with the instructions to link a large digitally-stored vocabulary to unique key images on the flat panel of the speech aid, thence controlling a speech synthesiser. Within the parameters of this basic format, various alternatives were explored. Some included alphanumeric key symbols whilst others allowed full-size icons to be drawn freehand on the panel. The BUILDER program is capable of changing any individual pages presented to the user. The USER program may have a softkey or remote switch input.

A number of Windows-based models have been produced. Prototype versions offer the speech therapist the choice of setting up a virtually unlimited number of speech keys dependent on specific patient needs. The strength of the original design flexibility has opened the way for more complex interactive models which still make use of the BUILDER/USER concept. However, instead of merely following a conceptual path to a speech output, partial reinforcement can be offered in the shape of visual and audio feedback. This permits the therapist to design repetitive training programs for specific utterances making use of stored high quality speech. Work is currently being done on more complex techniques which will be particularly useful for patients requiring language skills. In this mode the system will initially be controlled by a speech therapist. For example, an iconic symbol could be displayed as a trigger for an utterance from the patient [3]. If necessary, partial utterance delivery may be given by the system as the patient seeks to produce the required speech.

References

1 Musselwhite, C.R: Using Scanning Switches With The Micro-computer', Communication Outlook, Vol. 10, No. 1. July 1988,pp 12 - 13.

2 Newell, A.F: 'How Can We Develop Better Communications Aids?' Augmentative and Alternative Communication, Vol 3, No 1, March 1987, pp 36 - 40.

3 Bailey, S: 'Blissymbolics and aphasia therapy: a case study, Aphasia Therapy; 1983, pp 178 - 186.

Coping with a Multitude of Users in Multimedia

D. CAMERON
School of Occupational Therapy, Curtin University of Technology, Perth, Western Australia, 6008

Reduction in class contact time devoted to recreation and leisure studies led to the development of an interactive multimedia database designed to partially compensate. *Recreation Perth* was planned initially as a resource to assist therapy students select recreational activities to suit clients' needs.As this project has developed, secondary benefits have become apparent which may outweigh the initial objectives. The exercise has provided unique learning experiences for students, many of whom have had limited previous experience with computers. Additionally, the database's design allows information to be presented in different ways using a range of media options, thereby allowing it to be employed by therapists and accessed independently by clients with special needs.

One of the initial concerns was that occupational therapy students, in the past, had displayed limited computer knowledge and there may be hesitancy or resistance to use of the database. During 1992, the author interviewed new undergraduates to establish the extent of their previous computer experience. The results found that although only two students out of a population of 53 had no personal computer experience only one, from this mainly female population, had completed a computing unit in final year of high school (Hattie and Fitzgerald, 1987). Almost all considered their level of "expertise" and knowledge very limited. This restricted degree of computer awareness and competency raised some concern, not only to the success of *Recreation Perth*, but to the ability of qualifying therapists to serve the needs of their clients adequately. The undergraduate program currently has little formal call for student involvement with computers, but this technology can offer so much to therapist's clients, particularly those with developmental and physical disabilities (Niemec and Walberg, 1992).

To address these concerns, emphasis was placed on designing the database to make it appealing to the users with appropriate navigation aids based on proven educational principles. Students have been involved significantly in the design and development of this project. For instance, students working individually or in groups, have undertaken research seeking appropriate design requirements for adaptations of the database to suit the needs of both therapists and client groups. Over 100 students have provided the population in trials to test the effectiveness of screen designs, navigation aids and help features (Lee and Cameron,1994). To date, over 260 students have been involved in researching information on recreational pursuits and reporting their findings in a range of media formats for database entry. This exercise is also providing a useful means to assess student performance. Working in small groups, students are required to produce a comprehensive proposal on the recreational activity they intend researching. On approval, they commence a structured procedure which includes each student participating in the activity, collecting/producing information and presenting their findings to their peers and submitting a formal report which includes a critical analysis of the activity.

Last year, *Recreation Perth* won a federal government Teaching and Learning award to enhance its design and explore methods of assisting new users to interactive multimedia. This funding has allowed: ease of data loading features to be designed; improved graphics; optional Help and Guide features researched; preparation for CD-ROM mastering. A strategy has evolved in the design and development of this multimedia database which is currently being applied and tested in a new multimedia project. This is a self-paced learning package covering standardised tests for assessing clients suitability for particular vocational occupations. This strategy involves: consideration for user's memory limitation; overviews for new users; quality graphical interface; "human" contact with help features; and last but not least user involvement in design and development.

References

Hattie, T., & Fitzgerald, D. (1987). Sex Differences in Attitudes, Achievement, and Use of Computers. *Australian Journal of Education, 31*(1), 3-26.

Lee, H. & Cameron, D.F. (1994). Strategies to Provide Pathways for Learners in a Computer-Managed Learning Environment. In *Proceedings of the International Multimedia Symposium*, Perth, Australia

Niemec, R.P. & Walberg, H.J. (1992). The Effects of Computers on Learning, *International Journal of Educational Research, 17*(1), 99-108.

An Intelligent System for the Education of Non-Linear Dynamical Systems

OSCAR CASTILLO
Department of Computer Science
Instituto Tecnologico de Tijuana

PATRICIA MELIN
CETYS Tijuana
Department of Physics

P.O. Box 4207, Chula Vista CA, 91909, USA

We describe a computer program that can be considered an intelligent system for a domain of mathematics. The domain is known as "Dynamical Systems Theory". This domain is quite complex since it has been shown that even a very simple "dynamical system" can exhibit a behavior known as "chaos" (Devaney,R.,1989). The computer program uses Artificial Intelligence (AI) techniques to model a human expert in the process of teaching "Dynamical Systems Theory". The computer program contains the knowledge of the human experts expressed as rules and facts (knowledge base), and uses as a tool of reasoning the inference engine of the PROLOG programming language (Covington,M.A.,1988). As a result of efficiently using AI methodology for the education of Dynamical Systems Theory, we obtain an intelligent system that can be used as a tool to teach this complicated and relatively new area of mathematics To our knowledge this is the first intelligent system developed for the domain of Dynamical Systems.

Given a Dynamical System (DS) the computer program "knows" what numerical calculations have to be done to obtain all the possible behaviors of the system for different parameter intervals. For this the program has two modules. The first module has all of the knowledge about the theory of DS including the methods needed to identify the possible behaviors. The second module has all of the numerical methods needed to calculate the solution of the DS. The intelligent system also has a data base of examples of known DS that a student of this area of mathematics has to learn to get a deeper understanding of the theory

The intelligent system has been tested with encouraging results with a group of students of the "Program for Mathematics and Physics" (PROMYF) of CETYS University in Tijuana, Mexico. The PROMYF program is aimed at young students from Universities, to motivate the learning of mathematics and physics through the use of computers in the education, and also through teaching exciting new areas of mathematics (like Dynamical Systems, Chaos and Fractals), and their applications to physics and engineering. The PROMYF program is organized by the School of Engineering of CETYS University each summer since 1992 and it is aimed at University students of the Tijuana area in Mexico. The goal is that this intelligent system can become a useful tool in teaching this new area of mathematics.

References

Covington,M.A., Nute,D., Vellino,A., (1988). "Prolog Programming in Depth", Scott Foresman and Co. Computer Books.
Devaney,R., (1989). "An Introduction to Chaotic Dynamical Systems", Addison Wesley Publishing Company.
Sterling,L., Shapiro,E., (1987). "The Art of Prolog", MIT Press Cambridge Mass.

Evaluation of a Multimedium Learning System with Social Context

Huey-Wen Chou
Associate Professor, Department of Information Management
National Central University, Chungli, Taiwan, R.O.C.

INTRODUCTION

Multimedia applications, with its rapid growth of related technology and dropping of hardware prices, is receiving considerable worldwide attention by educators within the field of instructional technology. By accessing and integrating versatile information, such as video, sound, image, and text, multimedia applications in education afford a great potential in improving teaching quality.

As the computer hardware costs dropping down with functions improving impressively, the computer assisted learning (CAL) has brought a dramatic potential. Nevertheless, the CAL environment causes other problems. One of the major disadvantages in conventional CAL environment is the lack of social interaction. It was argued that learning environments with various degrees of social context would affect learners' motives, which, in turn, will have differential effects on learners' performance (Berlyne, 1978). In the present study, the effects of three learning environments with various degrees of social exposure on learning performance were examined.

DESIGN METHODS

A total of four pairs of comparison on learning effects were examined in this study:

self-directed learning vs. cooperative learning,
self-directed learning vs. competitive learning, and,
cooperative learning vs. competitive learning.

In cooperative and competitive groups, subjects know and can see each others during the experiment, whereas in self-directed learning group, subjects took an independent study via computer. It is assumed that subjects all acquire with appropriate entry ability levels to learn. Due to the environmental constraints, the experiment was held in two continuous days, only six subjects took the experiment on the same day.

DISCUSSIONS AND CONCLUSION

Both the pre-test and post-test scores in three groups did not show a significant differences. That means all subjects were with the same entry ability level, and there were no significant treatment effects. It was also found that more subjects preferred to learn in the competitive environment than in the other two environments. In addition, it was found from survey that most subjects would like to learn with companions who are better than himself/herself. This is an interesting finding which may be unique to Chinese culture. It is then suggested for future study to do homogeneous-ability grouping within treatments for maintaining subjects' optimal motivation.

The other important finding is that in the cooperative learning environment, subjects seldom exchange opinions with their partners by typing in text to the communication window. Another finding is that not many subjects in the competitive group sensed the pressure of time limit. In other words, subjects in the competitive and cooperative learning environments were actually in the self-directed learning environment. It is no wonder there existed no significant difference in post-test scores.

Researches of the different learning models' effects on learning performance and on motivation have been a lot, the results are not consistent - some are even contradictory. How to choose representative learning models with significant factors operating inside is still a big challenge. The study serves as a pilot one for future deliberation. By building up such a distributed social learning environment with multimedia support, it is the researcher's hope to provide a prototype for futuristic learning environment and bring impacts on education revolution in the coming centuries.

REFERENCES

Berlyne, D. E. (1978). Curiosity and learning. *Motivation and Emotion*, 2, 97-175.

Does Maths CAL Help Third-World Students ?

N.L.O. COWLEY, G.DE V. and M.M. DE KOCK, and D.J. KRIEL
Dept. of Computer Science,
University of Port Elizabeth (UPE), P.O. Box 1600, Port Elizabeth. 6000. South Africa

The project aimed to assess the effectiveness of computer assisted learning (CAL) in supporting the mathematics education of students from the disadvantaged communities (mainly Black) at a supplementary education Centre, called Ethembeni ('Place of Hope'). For various reasons the quality of education in most Black schools in South Africa has been and still is very poor. The Centre is building the learning culture essential for academic progress and its students are highly motivated, self-disciplined, and possessed of a desire to learn. The Centre is mainly funded by business. In order to help larger numbers of students the Centre considered CAL and approached us to advise them on starting a CAL facility, assist in the selection of a suitable systems, monitor and support the implementation of a CAL facility, and provide ongoing support thereafter.

A team of researchers from the Depts of Computer Science and Mathematics at UPE evaluated two candidate maths CAL systems experimentally. One of the systems is a locally-developed specialist maths CAL system and the other an internationally known content-generic system. Both systems provided course management facilities. The 140 11th grade students enrolled for a maths course at Ethembeni, were used for the project. From this population, 3 groups of 40 students each was chosen by stratified sampling; one each for the two CAL systems mentioned and one as a control group. The stratification was based on a standard pretest score, gender, age and school.

All groups attended maths classes at Ethembeni for the duration of the course (7 weeks). The control group received no CAL. One experimental group attended the CAL lessons on Friday afternoons and the other on Saturday mornings for a session of three hours. The same topics were covered in both groups. surds, and The CAL systems presented the students with problems, marked their answers, recorded their responses and provided help when necessary, BUT the students worked out the problems themselves on paper. Thus they remained independent problem-solvers, capable of functioning without computer support.. We also played music while the students worked. They enjoyed this very much and it did not seem to affect their ability to concentrate. They solved math problems while keeping time to the music.

Finally the student attitudes towards the hardware, CAL systems and courseware used and the project administration were evaluated by means of a questionnaire. Learning outcomes were evaluated by means of an end-of-course test written by all the students. This test was set and marked by the maths teachers at Ethembeni and moderated by the research team.

Both experimental groups learned to use the interfaces of the CAL systems surprisingly quickly, becoming competent after one three-hour session despite the fact that the students were initially all computer-naive. The students proceeded at their own individualised learning tempos. The bright students rushed ahead, while slower learners were not demoralised by being left behind by the others. The students experienced the CAL systems as non-threatening and relaxing to work with, and had very positive attitudes towards the project and the use of CAL in teaching mathematics. We were astonished to observe that the students were able to concentrate and work for three hours at a time.

Student responses were positive to all the closed response questions. They did not mind doing the course in English, although it was their second or third language. The major sources of complaints were the sandwiches they received and the impatient bus drivers!

The statistical results showed that the null hypothesis, viz that the use of CAL does not improve the mathematics competence of the students at Ethembeni, can be rejected with a confidence limit of 95% in the case of one group. Although the other group also showed improvement it was not significant. A detailed report of the project and the findings is available (internal report no. 94/02). The team has learned a lot about the practical aspects of running a CAL facility, how to study the interactions of students with CAL software and the nature and needs of disadvantaged learners. After the experiment ended, it was decided in consultation with Ethembeni to continue and expand the program with the objective of educational upliftment of both students and teachers.

Co-Learning at a distance : FIRST trial of an integrated learning environment

P. CROISY, D. CLEMENT, L. BARME
Laboratoire TRIGONE - Institut CUEEP
Bat C6 - Cité Scientifique - 59655 VILLENEUVE D'ASCQ Cédex - FRANCE
Phone : +33.20.43.49.21 - Fax : +33.20.43.69.67 - E-mail : pascal@univ-Lille1.fr

In our educational Institute, we are trying to enhance Open Learning [JOHN 93]. We take into account the idea that learning is a social process [RESN 91]. It is based upon a certain type of "mediation", as Vygotsky said.

So, we propose two types of mediation, through two classes of resource :

- human resources : teacher, tutor, expert and learners together whose relations are based on the "socio-cognitive conflict" that W. Doise defines as a constructive interaction between peers [DOIS 84];
- technical resources : we have built a multimedia environment for education within the European Delta Co-Learn project [DERY 93][VIEV 93].

One of the four prototypes, named RTMConf, provides a group of students with an integrated learning environment. The learning activity is supported by a platform for the collaborative work (CSCW). The system is designed for a small group of participants using the metaphor of a virtual meeting classroom.

The experiment

The students have to solve together a mathematical problem using the different tools of the RTMConf : the public window, the telepointer, the opinion collector, the time manager, the audio tool and the welcomer tool. They communicate together through PC stations under Windows 3.1. These stations are connected to a server by ISDN (Integrated Services Data Network) and include an audio channel. They were four students sites (with four students in each sites) and a teacher site.

Results

The experimentation validated the physical architecture in terms of performance and usability of the different used networks and tools. The metaphor also seems to be correct and the proposed tools effectively help the communication between learners. The teacher must help the learners to use the right one, adapted to their current task.

The audio channel is necessary to establish a good and rapid of the task and the coordination.

But for the usability of the system, we have seen that to put more than two students by site was an error. The teachers need to manage simultaneously a great number of information (technical and pedagogical). They should also change their usual mode of teaching to benefit from the system. So, we have to plan a good formation of teachers who will conduct some Co-Learn session.

Bibliography

[DERY 92] A.C. Derycke et al., Representation Models for Collaborative Educational Situations and Collaborative Learning Activities, Deliverable 4 of the DELTA CO-LEARN Project, 1992.

[DOIS 84] W.Doise, G.Mugny, The social development of the intellect, Oxford, England, Pergamon Press, 1984.

[JOHN 93] D.W. Johnson, R.T. Johnson, Learning Together and Alone, Cooperative, Competitive, and Individualistic Learning, Third Edition.

[RESN 91] L.B. Resnick, J.M. Levine, S.D. Teasle, Perspectives on socially shared cognition, APA Washington, DC.

[VIEV 93] C. Viéville, A.R. Kaye, F. Hoogstoel, A Collaborative Learning Activity Server as a Heart of a Communication System, Telepresence'93, First International Conference in Technologies and Theories for Human Co-Operation, Collaboration, Coordination, Lille, March 1993

Mathematica as a Conjecture Making and a Multimedia Tool

TILAK DE ALWIS
Department of Mathematics
Southeastern Louisiana University, Hammond, LA 70402, USA

Introduction

Today many university instructors across the world are integrating computer algebra systems (CAS) into their mathematics curriculums. One of their goals is to make the subject matter more appealing to the student. The wide availability of the CAS has indeed revitalized the teaching of mathematics as well as the research. If used the right way, it has a big potential of exploring every corner of a curious student's mind, giving him ample opportunities to form, test, and prove mathematical conjectures (de Alwis, 1993).

The popular CAS include, Mathematica, Maple, Derive, MacSyma, Theorist, Reduce and muMath. However, in this paper we will concentrate only on Mathematica. It is a CAS that can be used as a numerical or symbolic calculator, a tool for graphing, or as a visualization system to analyze data. One of the biggest strengths of Mathematica is its powerful built-in programming language. This provides an ideal tool for forming and testing conjectures of mathematical or physical phenomena. By combining the Mathematica programming language with its animation and sound capabilities, one can make mathematics spring into life! For example, in the subsequent sections, we will observe projectiles moving through the sky emitting various sounds and dynamic models for world population growth. These are excellent examples on how to use Mathematica as a multimedia studio. Not only that, we will also discover some useful theorems in the process. Our students will definitely be inspired by mathematics taught and presented this way. Consequently, they will be able to understand and appreciate the subject more.

What we have used is Mathematica standard version 2.0 on a Macintosh IIfx platform running at a clock speed of 40 MHz. Some good references on Mathematica are (Wagon, 1991) and (Wolfram, 1991).

The Motion of a Projectile with and without Air Resistance

Consider the motion of a projectile fired from the origin (0,0) with a velocity v ft/sec at an angle θ radians with the horizontal. Let (x,y) be its position at time t seconds and g be the acceleration due to gravity in ft/sec^2. Then in the absence of air resistance, its equations of motion are given by $x = vCos(\theta)t$ and $y = vSin(\theta)t - gt^2/2$ (Symon, 1971). By eliminating t between the above two equations, one can obtain the equation of the trajectory of the motion $y(x) = xTan(\theta) - gx^2/(2v^2Cos^2(\theta))$, which is a parabola. In a typical calculus class one usually does not consider the motion with air resistance because of the complexity of the equations of motion. However, with a powerful CAS like Mathematica, students can analyze this motion as well. Let m be the mass of the projectile. As given in (Symon, 1971), one popular model is to assume that at any time t, the frictional force due to air is directly proportional to the velocity at that time. By denoting this constant of proportionality by b, one can show that the equations of motion are given by

$$x = \frac{mvCos(\theta)}{b}(1 - e^{-\frac{bt}{m}}) \quad \text{and} \quad y = \left(\frac{m^2g}{b^2} + \frac{mvSin(\theta)}{b}\right)(1 - e^{-\frac{bt}{m}}) - \frac{mgt}{b}.$$

Eliminating t between these last two equations and denoting y by $y_{air}(x)$, one obtains the trajectory of the motion with air resistance

$$y_{air}(x) = \left(\frac{mg}{bvCos(\theta)} + Tan(\theta) \right) x - \frac{m^2 g}{b^2} ln \left(\frac{mvCos(\theta)}{mvCos(\theta) - bx} \right)$$

Using these equations, one can write a Mathematica program to simulate the motion of a projectile with and without air resistance in the same set of axes. In the actual presentation, one can observe the motion of two projectiles, one in solid red (without air resistance) and the other in dashed blue (with air resistance). On the top of each graph, one can see the current time and the position of each projectile. Not only that, accompanied with every movement of the projectile, one can also hear an interesting sound. At the very bottom, the maximum heights and the ranges of the projectiles are displayed. One can export a few frames of this animation to a graphic program such as Canvas. Within Canvas, one can start editing these frames. One can also import images from popular Clip Art packages and then do the final editing within Canvas. Then import the final edited frames back into Mathematica.

World Population Growth - Exponential and Logistic Models

Let us consider two models for world population growth. According to the exponential model, one assumes that the rate of growth of a population is directly proportional to the number of individuals in the population at that time (Ross, 1989). Let t denote the time in years, and $P(t)$ denote the population at time t. This means that $dP(t)/dt = k_1 P(t)$ for some constant k_1. It can be easily verified that the solution of this differential equation is $P(t) = P(0)e^{k_1 t}$ where $P(0)$ denotes the initial population. The exponential model seems to work very well over a short period of time for human populations and also for certain mammalian species under certain conditions. However, it is unrealistic for longer periods of time. The logistic model is proven to be more satisfactory for restricted environments with a limited amount of food supply. In such an environment, let M denote the maximal population. The logistic model assumes that the growth rate $dP(t)/dt$ is directly proportional to $P(t)(M - P(t))$ (Ross, 1989). Hence one can write that $dP(t)/dt = k_2 P(t)(M - P(t))$ for some constant k_2. One can also show that the solution of this differential equation is given by $P(t) = P(0)M/[P(0) + (M - P(0))e^{-k_2 M t}]$. This equation implies that $\lim_{t \to \infty} P(t) = M$. Therefore, unlike the exponential model, the logistic model is more realistic.

Based on the above equations, one can write a Mathematica program to simulate the world population growth. In the actual presentation, one can observe two graphs climbing up as the time t increases, one in dashed red representing the exponential model and the other in solid green representing the logistic model. The current year and the total populations predicted by the models are also displayed. One can also use the graphic program Canvas and a Clip Art package to create images of a big metropolis and import them back into our animation. This is a prime example on how to use Mathematica as a multimedia studio.

References

de Alwis, T. (1993). Mathematica and the Power Method. *International Journal of Mathematics Education in Science and Technology*, 24 (6), 813-824.

Ross, S. (1989). *Introduction to Ordinary Differential Equations.* New York, NY: John Wiley and Sons.

Symon, K. R. (1971). *Mechanics.* Redwood City, CA: Addison-Wesley.

Wagon, S. (1991). *Mathematica in Action.* New York, NY: W. H. Freeman.

Wolfram, S. (1991). *Mathematica: A System for Doing Mathematics by Computer.* Redwood City, CA: Addison-Wesley.

Acknowledgements

The author wishes to thank Mr. James O. Morgan for his valuable suggestions.

An adaptive questionnaire based upon an automatic knowledge assessment tool

MICHEL C. DESMARAIS JIMING LIU
Centre de recherche informatique de Montréal
1801 ave. McGill College, bureau 800, Montréal, Québec, Canada H3A 2N4

One of the most praised qualities of computer assisted training (CAT) is its ability to deliver individualized training. In CAT, the pace of individual learning is not determined by the learning group's average, or maximum, or minimum learning pace. It adapts to the individual's capabilities. Even the content can be tailored to the individual's background knowledge and personal training needs.

Current state-of-the art CAT systems support individualization of the training process by allowing free or flexible navigation within the training content, as opposed to a rigid linear content presentation controlled by the computer. However, this flexibility imposes an additional burden on both the instructional designer and the learner. For example, the instructor must decide the appropriate sequencing of exercises according to the successes and failures of the learner. Or the learner must explore the whole training content and decide what is appropriate.

We present a method for reducing the effort and complexity of designing and delivering individualized training based upon the automatic assessment of the learner's expertise. The method induces, from emprical data, the learning order of knowlege units (KU—they can represent mastery of concepts or skills). This ordering is thereafter used to infer an individual's knowledge state (see Falmagne *et al.*, 1990, for the cognitive foundations of this method). Through a sampling of a person's knowledge state, the system assesses the probability that this person knows any given KU. The application of such method to adaptive CAT systems are numerous.

We built a system for an adaptive questionnaire based upon this method. A question represents a KU. Every time a user answers a question, the probability of correctly answering each other question is updated automatically, according to a success or a failure, based upon the empirically derived ordering of questions (which constitutes a *partial order*). The ability to use this information for adaptive purposes is shown in a mode where the system chooses the most informative question (based upon a question's expected value of entropy reduction given an answer). If the user correctly answers the questions chosen by the system, the degree of question difficulty tends to increase gradually. On the contrary, if the user fails systematically, the degree of difficulty will tend to decrease. For example, we constructed an adaptive questionnaire on UNIX. The automatic assessment module was initialized with data from 20 subjects. The following sequences represent the mean failure rate of questions presented to the user for successive successes and failures respectively:

Questions' mean failure rate for consecutive successes: 33%, 48%, 67%, 57%, 81%, 76%, ...
Questions' mean failure rate for consecutive failures: 33%, 48%, 24%, 38%, 24%, 24%, ...

Moreover, the user's *global score* will rapidly rise with the above successes and drop with failures:

User's global score with consecutive successes: 61%, 69%, 72%, 77%, 81%, 84%, ...
User's global score with consecutive failures: 61%, 54%, 51%, 49%, 47%, 45%, ...

Without the inference power from the automatic assessment, the resulting scores at the end of the above sequences of questions would correspond to 64% instead of 84% for successes, and 58% instead of 45% for failures.

Systematic validations of the method for correctly assessing an individual's knowledge state were performed in a number of different experiments on the knowledge of WordPerfect™ text-editing commands (Desmarais et al., 1993; Desmarais and Liu, 1993). These results showed that, for example, a close to perfect knowledge assessment was inferred after sampling 70% of a subject's knowledge state, and that sampling 50% of the knowledge state would reduce the standard error score of estimates to about half of the error score without inferences. The method thus successfully reduced the number of questions that needed to be asked to assess a subject's score.

References

Desmarais, M. C., Giroux, L., and Larochelle, S. (1993). An advice-giving interface based on plan-recognition and user knowledge assessment. *International Journal of Man-Machine Studies*. In press.

Desmarais, M. C. and Liu, J. (1993). Exploring the applications of user-expertise assessment for intelligent interfaces. In *InterCHI'93, Bridges between worlds*, pages 308–313.

Falmagne, J.-C., Doignon, J.-P., Koppen, M., Villano, M., and Johannesen, L. (1990). Introduction to knowledge spaces: how to build, test and search them. *Psychological Review*, 97(2):201–224.

Helping students get unstuck

Richard Feifer
USIU, 10455 Pomerado Road, San Diego, CA 92131

Douglas Holyoak
Center for Professional Education
Arthur Andersen & Co., 1405 North Fifth Ave, St. Charles, IL 60174

When students attempt to learn by doing, they will get stuck. If a human guide is available, the student can receive the support he or she needs to get moving again. We call this kind of support a *procedural nudge*. A student who is learning by doing in a computer-based environment has an even greater need for a procedural nudge. For maximum learning to take place, the student should receive just enough of a push or guidance to get past the sticking point without being explicitly told the answer. In addition to providing the right kind of help, designers should consider the personality traits students may attribute to the tutor. A clear, consistent personality enables the learner to appropriately interpret and use the procedural nudge. We have observed the following five human personality types providing procedural nudges within computer-based learning environments.

The **Eager Expert** understands the domain, knows the student's problem, is smart, and, because it doesn't have the patience to guide the student to the answer, it simply gives the student the answer. While it is probably the most typical kind of assistance currently available within educational software, it rarely results in learning.

A **Gaming Genius** also understands the domain, knows the problem, is smart, but seldom gives direct answers. This personality provides the student with vague hints and ambiguous answers. The Gaming Genius is usually implemented to give just enough information to get the student moving again. The student, however, perceives that the Gaming Genius is toying with the student and enjoys seeing the student struggle.

The **Fallible Colleague** may or may not understand the domain or the problem, requiring the student to carefully evaluate any advice he or she receives. This personality has good intentions, offering many solutions or answers, yet some answers are wrong. One pedagogic advantage is that the tutor can give the answer without fear of giving it away, since the student will not be completely confident that the answer is correct.

The **Clever Confidant** does not understand the domain nor does it know the problem, but it is smart. The Clever Confidant knows how to reason about problems in general and ask questions the student should ask him or herself. The advantage here is that the Clever Confidant begins to model the behavior the student should be exhibiting in the problem-solving process.

A **Blind Brain** understands the domain, is smart, but does not know the student's problem. With this personality the student is forced to conceptualize his or her problem before the tutor can help. Any answer the Blind Brain provides is only as good as the conceptualization the student provides. Unlike the Fallible Colleague, the appropriateness of the advice is not random, but dependent on the student's ability to understand and communicate the problem.

All of these personalities, with the exception of the Eager Expert, can provide a procedural nudge. Assuming the personalities are implemented in a clear and consistent manner, each will encourage the learner to develop different problem solving skills: Gaming Genius - making inferences; Fallible Colleague - judging the validity of data; Clever Confidant - asking oneself the right kind of questions; Blind Brain - conceptualizing the problem. Both Clever Confidant and Blind Brain have the additional advantage of accomplishing these pedagogical goals without annoying the learner.

An AI based programming environment for the learning of programming

Paola Forcheri and Maria Teresa Molfino

Istituto per la Matematica Applicata del Consiglio Nazionale delle Ricerche
Via De Marini 6, 16149 Genova - ITALY Email: {FORCHERI,MOLFINO}@IMAGE.GE.CNR.IT

The variety of programming paradigms widely used in practical situations stresses the importance of building computing environments aimed at helping students acquire the capability to computationally model problems independently of a specific language. Moreover, if such environments are endowed with the capability of automatically producing code in different languages (i.e. imperative, logic, functional, etc..), students are operatively motivated to study modelling problems, and are provided with a tool for rapid prototyping on different programming schemes. On this basis we have designed a system, called MEMO-II, intended to be used in the teaching/learning of programming with University students. A prototype version of the system runs on IBM/486 under SCO-Unix operating system.

MEMO-II is a programming environment which allows the user to build programs from formal specifications via interaction with the system. The process of generating a program from a specification is carried out by the system.

To help students understand the different processes intervening in the building of programs, that is how a model is built and modified, how interesting conjectures are made and verified, how a representation can be derived, MEMO-II places a series of facilities at the disposal of the user. Via the facilities, the user autonomously decides about the specification to be explored, formulates properties which must be verified and the knowledge to be used in the proof, establishes which links with mathematical structures should be analysed, decides which tests must be carried out. The system helps the user in this activity by guiding him in the description of the specification, by verifying its adequacy with respect to feasibility, by giving proof of the properties indicated by the student and by deriving programs from specifications.

The process is carried out interactively; the facilities of the system are accessed through commands. An algebraic specification oriented language, a validator of specification, an automatic translator and a theorem prover are the core of the system. A broad analysis of the technical organisation of the system and its components can be found in [Antoy et al. 1993].

From a pedagogical point of view, the work is based on three main observations: it is commonly acknowledged that we need efficient ways to teach abstraction; teaching modelling contextually with a language makes it difficult to understand the difference between the conceptual and technical processes leading to solve a problem computationally; the amount of time and effort needed to build computer systems devoted to teaching problem solving suggest making them applicable in a variety of educational contexts. To overcome these difficulties, we propose using systems endowed with the following features: 1) to support a variety of programming languages in order to be applied to several educational contexts; 2) to place at students' disposal a series of tools for the development and testing of formal specifications, independently of a particular programming paradigm; 3) to grant the user facilities for producing effective programs focusing his attention on the cognitive aspects of programming rather than on the technical ones.

[Antoy et al. 1993] Antoy, S., Forcheri, P., Molfino, M.T., Schenone C. (1993). A uniform approach to deduction and automatic implementation. In *Design and implementation of symbolic computation systems*, Fitch J. (ed.), Springer-Verlag, 132-141.

HCDE: A Hypermedia _ Based Courseware Developing Environment

LIU GANNA

Dept. of Computer Science and Engineering Research Center of CAI
Xi'an Jiaotong University, SHaanxi, China, 710049

The Hypertext and multimedia information is merging into the hypermedia at the present. A hypermedia _ based courseware developing environment, HCDE, has been developed and implemented at Xi'an Jiaotong University. The main functions and features of HCDE are as follows:

1. **HCDE provides the multimedia information interfaces in the low _ cost hardware environment**, so it can be popularized easily. These interfaces can link the hypertext nodes with the text, graphics / image, animation and the Chinese speech. They are constructed by software technology without speech card, video _ card and other special hardware.

2. **A novel "bus _ oriented" message driving control model in HAE is designed** for increasing the independence of modules, extensibility and maintainability. HCDE consists of the Hypermedia Authoring Environment (HAE) and a ToolBox which includes five special tools. HAE includes 12 object classes (or called modulus) to organize all forms of information into the card of hypertext and automatically generates a hypermedia courseware by defined 15 messages driving. The sending message and control transfer take place on the message queues, exchanging information between the modules occur indirectly.

3. **HCDE is a non _ programming opening environment** for teachers and allows them to write the courseware script in English or Chinese by simple authoring rules (only 16 control symbols as prefixes of the objects) rather than in programming way. The tools have friendly HCI and detailed help information. The efficiency of developing courseware can be raised tens times.

4. **HCDE supports courseware engineering, creates knowledge base of ITS and student model.** It also supports interactive teaching applications, instructional path control by logical lock. A good navigation algorithm is designed to avoided the 'lose — way' problem, in the meantime, deadlock problem has been solved.

5. **The object _ oriented method and self _ adaptive are adopted in HAE to realize the compatibility, tailorability and adaptivity of system.** A new and original speech signal compression algorithm ZBCOMP on Chinese Speech Synthesis System and the new ideas about generating animation are achieved. HCDE was developed in BORLAND C^{++} 3.0, running on IBM _ PC with 80x86 CPU, VGA, HD(>20M) and LOUDSPEAKER. The system owns Chinese character library, supports mouse and keyboard. It has been used to develop courseware, electrical document and knowledge base of ITS at more than ten universities.

References

H. maurer, Why Hypermedia System are Important.
In proceedings of ICCAL'92

F. Halase, Reflections on Notecards: Seven Issues for the Next Generation of Hypermadia Systems, *Communications of the ACM*, 31, 7, 836—850

B. La Passradiere, A CAL Environment for HyperCard,
In proceedings of ICCAL'89.

Italo De Diana & Paul Van Schaik · Couresware Engineering Outlined: an Overview of Some Research Issues,
ETTI 30,3 191—208

Mathematical Objects in a Visual Interactive Environment : MOVIE

J.N. GERS, A.C. DERYCKE
Laboratoire TRIGONE
CUEEP Institute, University of Lille I, 59650 Villeneuve d'Ascq, France
Alain.Derycke@univ-Lille1.fr

Cognitive and social sciences have pointed out some fondamental caracteristics on the nature of learning (Pea, 1992) :

- the development of intelligence is actively constructed by the individual,
- learning is situated in a community of practice,
- learning is a process of communication between human beings,
- learning aims to everyone's autonomy and cooperation.

According this results, a mediatized learning environment must provide two essential functionalities :

- it must provide the learners with an interactif and shared workspace. It is the field of the action and of the observation.
- it must enable and ease the conversation about the action in real time and afterward. It is the field of the communication and of the negociation of meaning.

We are developing such an environment, called MOVIE, for learning mathematics. MOVIE is a visual interactive environment for the calculus. The objects of MOVIE are all kinds of usual objects (price tags, gauges, clocks...) or mathematical objects (formulae, graphs, figures...) it is possible to calculate with.

Composed objects are built by copy and by paste of basic objects. For the calculus we use a visual formalism wich extends the algebraic coding to a third dimension, according to a spatial syntax.

MOVIE is thought from scratch for the collaborative work : the learners put on the shared workspace the objects they are appropriate to the studied problem. These one are given to everybody's action.

MOVIE is multimetaphors : the one of the direct manipulation and the one of the language coexist for the managing of the calculus.

MOVIE is multimedia : the change of standpoint on the calculus can enable cinematical animations.

MOVIE is multiagents : we can put agents in the environment. They relieve human operators in some tedious tasks.

We have developed a first prototype in Smalltalk 80 (Gers, 1993).

References

Bricker,W. (1992). Spatial Representation of Elementary Algebra *IEEE*.

Derycke,A.C. & Vieville,C. (April 24-27 1993). Real time Multimedia Conferencing System and Collaborative Learning. *NATO Advanced Research Workshop, UNED*, Segovia, Springer-Verlag. F.Verdejo et al (Editors).

Gers,J.N. (1993). Icones dynamiques et enseignement des mathematiques. *Environnements Interactifs d'Apprentissage avec Ordinateur*, M.Baron, R.Gras, J.F.Nicaud (editions Eyrolles).181-191.

Pea,R.D. & Gomez,L.M. (1992). Distributed Multimedia Learning Environments : Why and How ? *Interactive Learning Environments*, Vol.2, Issue(2), 73-109.

Tanaka,Y. (1990). A toolkit system for the synthesis and the management of active media objects. *Deductive and Object-Oriented Databases*, W.Kim, J.M.Nicolas, Nishio (Editors) Elsevier Sciences Publishers B.V. (North Holland), 259-277.

Glucomedia: How to integrate multimedia environments & ITSs through advisory strategies and learner modelling

M. GIARDINA, M. LAURIER, AND C. MEUNIER
GRAEMI, Département d'études en éducation et d'administration de l'éducation
Université de Montréal, C.P. 6128, Succ. A, Montréal, Québec, H3C 3J7, Québec, Canada

The combination of multimedia environments with ITS takes advantage of the various ways the information can be processed. The information is related to the facets of the content to be learnt but it is also related to the characteristics of the learner. Interaction is possible when the progression through the available network of information depends on what the system knows about the learner.

The concept of significant interaction

An intelligent system has the ability to reason about the direction which the learning interaction should take at any given moment. The accuracy of the model that is used to adapt the interaction depends on the quality of the inferences that can be made by the system regarding the learner's cognitive abilities, needs, believes and strategies. Significant interaction occurs when the information is interpretable by the learner in terms of meaningful units of information that can modify the learner's representations or motivate him/her to go further. The learner is thus participating in a multi-modal transaction defined as exchange cycles of information, more or less meaningful, to generate significance or understanding of a phenomenon or a particular state (Merrill, Li Zhongmin & Jones Mark, 1991). The main pedagogical issue is to give the learner more control over its interaction with the system and, at the same time, to allow the system to adapt its interventions during the learning session.

In the design of environments based on significant interaction, three dimensions are interwined:
- **The intervention level**: The systems interacts significantly with the learner at different levels: *perceptive*, *transactional*, *cognitive*, *pedagogical* and *evaluative*.
- **The structural components**: *Objectives* describe expected *representations* to be set up though the use of various *strategies*.
- **The quality criteria**: Every component, at every level, should be *clear*, *coherent* and *relevant*; it includes some *analogy* (denotative and connotative), some *redundancy* (density of information) and some degree of learner's *control*.

An example: GLUCOMEDIA

The main objective of *Glucomedia* is to motivate, inform and teach patients who suffer from diabetes to improve their control over the desease. The system presents day life situations which call for a decision from the patient. At any time, according to the evolution of patient's knowledge, *Glucomedia* can complexify the situation presented, incorporating new elements that may force the patient to revise his/her previous decision or to make new ones. The intelligent learning environment is based on two qualitative models: free exploration and apprenticeship. The multimedia components of *Glucomedia* have been integrated in the knowledge base. The multimedia functions can be activated by the advisor in conjunction with the evolution of the learner's knowledge.

Reference

Merrill, D., Li Zhongmin & Jones Mark, K. (1991) Instructional Transaction Theory: An Introduction. *Educational Technology* 30(2), 7-14.

The lessons of user trials on the development of a Hypermedia Learning Environment.

Simon Gill, and David Wright
Department of Design, Brunel University, Surrey, England

We have been developing a Hypermedia learning environment based on the Constructivist model of teaching for several years now (Gill 93). From our own observations and also those in other institutions we recognise that many students hold deep misconceptions, and incorrect models of many physical laws, such as Newton's Laws of Motion (Osborne & Freyberg, 85). We have implemented the constructivist model within a HyperCard™ based system which allows the user to branch and follow non-linear routes of learning with extensive multimedia support for this particular topic. We have been experimenting with these principles for a number of years now, and have developed four discrete systems to date. We are committed to an on-going testing of the system, in order to gain important feedback, which is tremendously important in educational software development. Our systems have been successfully tested at the Department of Design, Brunel University; Fachhochschule Hannover, Germany; the Rivers School, Massachusetts, USA; and the Department of Education, University of South Dakota, USA. The age of the students range from 14 years to 24 years, with varying backgrounds, which has provided us with interesting data.

Results

Contrary to some of our earlier assumptions, the models held by the students are not self standing ideas that they apply universally to many different phenomena, but are a kind of sub-conscious fuzziness, which consists of some formal scientific explanation. A student will pick certain terms and definitions from the correct concept, and then mix it with their own wild inaccuracies to form their explanation. Students have a tendency to cling to key words like 'resistance' and cheat the system looking for correct answers, which inevitably hold recognisable key words. Many of the younger students treated the system as a game, always wanting to know if they were right or wrong. However the older students are quite happy with the informal approach, taking much more time, and was best summarised by one student as "doesn't say you are right or wrong, you have to think and understand it". The use of audio for signifying key interactions was greeted with a positive response. However the use of spoken human directions for certain activities was found to be ineffective and often annoying. Students were asked if they would like to have the texts read back to them and the answer was a resounding 'yes'. Such a feature has been easy to implement with the new Apple PlainTalk extension. It must be noted that the younger students had an aversion to reading much of the card texts. Each card held a reasonable amount of text, with up to 15 lines and 180 words (on a 640 x 480 pixel card, with 14pt Palatino). This is not a large amount of text compared to many systems, but it was often too much. Extreme cases saw users flick to a card, be presented with a lot of text, and then move quickly on to the next card without stopping to read the text. These students were often on 'the hunt' for the Quicktime movies and animations. Macromind Director™ animations and Quicktime™ video sequences were originally played automatically on the opening of each card. However the students where not happy with this as they wanted to have more control when the animation appeared, when it played, and when it stopped. By allowing the user to open the movie at their request we have been able to improve card to card times, and to display multiples of the same movie on screen, allowing the user to examine sequences more thoroughly. All the students tested where more than happy to open these movie objects, and would often spend several minutes watching and rewatching a short movie. Comparing the success of the movies with the lack of interest for text, it is quite clear which holds more captivation for the students. Newer versions of the system now contain much less text per card, with more text, diagrams and explanations in the Quicktime movies. This is perhaps not a major surprise to anyone who has observed teenagers play computer games and the amount of TV they watch. These factors are undoubtedly important to the success of learning systems and these trials have provided us with many important points.

References

Gill, S.H. (March 1993) Multimedia learning and the theories behind individual discovery in the *Proceedings of The 10th International Conference on Technology and Education*, Boston, USA, pp. 352-354.

Osborne, R and Freyberg, P. (1985) *Learning Science*. Heineman, Auckland.

A Cross-Domain Experiment in Case-Based Design Support: ArchieTutor

ASHOK GOEL
College of Computing, Georgia Institute of Technology, 801 Atlantic Drive, Atlanta, Georgia 30332, USA

ALI MALKAWI
College of Architecture, Georgia Institute of Technology, Atlanta, Georgia 30332, USA

MICHAEL PEARCE
Galaxy Scientific Corporation, 2130 Parklake Drive NE, Suite 325, Atlanta, Georgia 30345, USA

Over the last few years, our research group has developed a series of case-based design support systems. Each system in this series represents an experiment in combining case-based and multimedia technologies for supporting conceptual design in complex domains. In the Archie project (Pearce et al., 1992), we explored the use of past designs for aiding architects in designing new office buildings. In the AskJef project (Barber et al., 1992), we investigated the use of multiple types of knowledge, including past designs, for advising software engineers on the design of human-machine interfaces. The more recent ArchieTutor system represents a cross-domain experiment in using AskJef's framework for supporting design teaching in Archie's domain.

ArchieTutor operates in the domain of architecture, as does Archie. However, instead of supporting professional architects in solving complex design problems, our goal for ArchieTutor is to support design teaching in beginning architectural classes. More specifically, the system is intended to support design teaching in two ways. First, building on the results of AskJef, it is intended to support design teaching by helping beginning architectural students in understanding the nature of the design domain of office buildings, and the structure of design problems and solutions in the domain. Second, following the results of AskJef and using Archie as a base, it is intended to support design teaching by exposing students to some of the knowledge sources and skills useful in design generation and design critiquing.

This form of ``subversive tutoring" does not force the student to learn about the domain but invites the student to explore the domain in the context of solving a design problem, and presents knowledge in such a way that it is likely to be remembered. The most important features are:

- ArchieTutor makes the relationship between the different types of knowledge explicit; the student has access to a cross-indexed knowledge network (design cases, design principles and guidelines, and domain models) in the system's memory.
- ArchieTutor provides a vocabulary for cross-indexing design cases, principles, and domain models.
- Organization of ArchieTutor's interface strongly reflects the organization of knowledge in its memory.
- Graphical representation of the case allows the student to focus on a specific chunk of the design by zooming in, or look at the design as a whole by zooming out.

References

Barber, J., Bhatta, S., Goel, A., Jacobson, M., Pearce, M., Penberthy, L., Shankar, M., Simpson, R., and Stroulia, E. (1992). AskJef: Integration of Case-Based and Multimedia Technologies for Interface Design Support. *Proceedings of the Second International Conference on AI in Design*, Pittsburgh, June 1992, pp. 457-476.

Goel, A., Pearce, M., Malkawi, A., and Liu, K. (1993) A Cross-Domain Experiment in Case-Based Design Support: ArchieTutor. *Proceedings of the AAAI-93 Workshop on Case-Based Reasoning*, Washington, D. C., July 1993, pp. 111-117.

Pearce, M., Goel, A., Kolodner, J., Zimring, C., Sentosa, L., and Billington, R. (1992). Case-Based Design Support: A Case Study in Architectural Design. *IEEE Expert*, 7(5):14-20, October 1992.

Supporting the Learning of Recursion at a Distance

JIM E. GREER, GORDON I. MCCALLA, BOB PRICE, AND PETER HOLT

ARIES Laboratory, Department of Computational Science
University of Saskatchewan, Saskatoon, Canada S7N 0W0

Research in the area of artificial intelligence in education (AI-Ed) has made progress on three fronts. First, AI-Ed researchers have constructed discovery learning environments in which a learner can use creative play with the aim of achieving self-actuated goals. Second, AI-Ed researchers have explored reducing the complexity of learning by providing scaffolding that can slowly fade. Third, AI-Ed researchers have begun to understand how to individualize tutorial interaction so that the cognitive needs of the learner can be dynamically fulfilled. The key to this individualization is the ability to perform "student modelling" where learning states are diagnosed from learner behaviour and incorporated into a longer term model of the student.

We have developed two systems for supporting the learning recursion in LISP. PETAL (Bhuiyan, 1992) provides a scaffolding environment in which learners are supported in their *mental model*-level problem solving as they solve programming tasks. SCENT (McCalla et al., 1988) is able to diagnose strategic and logical errors in learners' completed programs and is able to provide English advice about their solutions. In this research we have shown how PETAL can be used "stand-alone" by learners at a distance to receive the local support they need as they develop solutions to their programming tasks, but, when they feel the need for individualized advice on their completed solutions, the learners can receive it remotely from a centrally running version of SCENT.

PETAL consists of three Programming Environment Tools (PETs) to support learners in the use of three mental models of recursion. As learners use a PET, they express mental model-level decisions by actions carried out in the PET. PETAL can generate runnable LISP code from the mental model-level descriptions the learners create. This helps learners to offload onto PETAL the necessity for dealing with the intricacies of LISP syntax and semantics, thus allowing them to concentrate on problem solving rather than coding. We have found that the PETs themselves serve as a powerful learning tool that scaffolds problem solving for novice programmers.

The SCENT advisor is a traditional intelligent tutoring system in that it builds an internalized model of the student programmer's strategy in order to comment upon it. It is non-traditional in its use of *case-based* and *granularity-based* reasoning methodologies to help it to build this model. At the heart of the SCENT advisor is a recognition engine that determines what strategies the learner has used in his or her LISP program. The recognition engine uses a model-based approach to recognition, comparing the learner's solution to models in a fixed strategy library. In our approach, we overcome fundamental problems of model-based recognition, especially problems in robustness, flexibility, and adaptability.

In order to connect PETAL and SCENT and to run multiple PETALs at the same time, it was necessary to convert SCENT into an advice server, managing advice requests from multiple PETAL clients. The protocol involves PETAL sending an advice request to SCENT and when the solution has been analysed, SCENT returns the advice. The learner then browses the advice in PETAL while SCENT proceeds to handle other requests.

We conducted an experiment with nine first-year university students using the PETAL/SCENT integrated system. The protocols confirmed that students would work for some time before requesting advice. During the exit interviews students were positive about PETAL but were split on the perceived value of SCENT. Some were frustrated because SCENT would not give specific instructions about how to fix their code but they appreciated the fact that it would show where their code was wrong and give hints about how to correct it.

This work integrates systems from two different streams of AI-Ed research to synergistically provide a truly useful learning environment in a cost-effective way. We believe that the PETAL/SCENT integrated system is a role model for linking scaffolding environments to a central AI-based tutoring system.

References

Bhuiyan, S.H. (1992) *Supporting Students in the Use of Mental Methods of Recursion* Ph.D. Thesis, Dept. of Computational Science, U. of Saskatchewan, Saskatoon, SK, Canada.

McCalla, G.I. & Greer, J.E. (1989) SCENT-3: An Architecture for Intelligent Advising in Problem Solving Domains. In C. Frasson and G. Gauthier (eds) *Intelligent Tutoring Systems*. Ablex, 140-161.

Training services in a cooperative environment

A. HERNANDEZ-DOMINGUEZ
A.P.I. Laboratory, Paul Sabatier - University of Toulouse III, France.
50 chemin de Maraîchers, 31062, Toulouse Cedex.

The environment considered is an integrated and multi-site environment which must provide technical support for information management (designing and implementation facilities, maintenance), for information transmission (computer networks, mailing service) and for information processing (tracing and evaluation services). Such a tele-teaching centre requires two categories of pedagogical resource sites : • the main training centre which organizes, manages and diffuses the training services (pedagogical bricks) and • the training centres (remote sites), which provide the learners with all necessary human and technical services (ACTS : Adapted Class Training Service and AGTS : Adapted Group Training Service) required for the instruction process.

Training services in each training centre are structured in three layers : support, global and cooperative layers. Each training service of any training site is designed using an object-oriented approach, it is based on functional and independent 'building blocks' (Natarajan, 1992).

Support layer represents the information, network, software and hardware platforms.

Global layer represents four types of SITB (Service Independent Training Building Block) : pedagogical service to manage apprenticeship of learner group, information service for supporting learning activities, evaluation service to evaluate a learner group, and cooperation service allowing the interaction between users (to exchange messages and pedagogical resources). Pedagogical service is represented by basic resources (multiple-choice-questions), isolated or grouped (questionnaire), which constitute pedagogical 'bricks' that can be teleloaded from each training centre. Information service is represented by knowledge source such as : lexical concepts of domain. Evaluation service is represented by multiple choice question and questionnaires

Cooperative layer is represented by two cooperative services : ACTS and AGTS. ACTS is managed by a class manager controlling (by control interactions with the corresponding group manager) an electronic class according to a class teaching strategy, each learner group is classified by a knowledge level ('novice', 'confirmation', 'deepen' and 'master'). Each AGTS is managed by a group manager controlling a particular knowledge level according to a group teaching strategy. The components of each AGTS are : • actors (teachers, learners and reusable SITBs), • roles (actor's behaviour), a set of roles has been identified in the ECOLE (European Collaborative Open Learning Environment) project (Eijkelenburg, 93) : speak, react, ask, answer, observe and work, in addition to that we consider inform, evaluate, and tutor, • activities, a learning activity allows to establish a learning situation according to roles played by actors, and • the manager of pedagogical situation execution, it must to manage the training sessions, through a series of pedagogical activities adapting each pedagogical situation to learner's needs (individual or in group) in a flexible training.

A multi-site environment such as SEMIEC (in french : 'Système Evolutif Multimédia Intégré d'Enseignement de la Comptabilité') based on the training architecture (Hernández, 1993) is being developed in the distance education context of apprenticeship of accountancy at IPST (the 'Institut de Promotion Supérieure du Travail' is a french institute for professional training).

References

Eijkelenburg V.K. (1993). ECOLE : Ecole as a Computer Supported Cooperative Learning Service. In *Proceedings of the IFIP TC. Third Teleteaching Conference*, August, Trondheim, Norway, 20-25.

Hernández, D.A. & Gouardères, G. & Descargues, R. (1993). Un environnement multi-sites pour la production et la consommation des ressources pédagogiques. In *Proceedings of "ico 93", 4th International Conference on Cognitive and Computer Sciences for Organisations*, May, Montreal, Canada.

Natarajan N., & Slawsky G. (1992). A framework architecture for information networks. *IEEE Communication Magazine*, Apr. 1992, 102-108, USA.

Fuzzy Techniques for Understanding Students' Solutions in an Intelligent Tutoring System

CH. HERZOG AND H. ZIERL
Technische Universität München, Institut für Informatik
D-80290 München, Germany

Synchronizing parallel processes is a difficult and complex domain. To offer students practical experiences in this field we developed SYPROS, an intelligent tutoring system. SYPROS uses tree-like structured goals and plans for the representation of its problem solving knowledge (cf. [HeGo93]). An important aim in developing SYPROS was to make it able to observe a student's problem solving process and to understand a student's solution. For this reason the goal-plan-tree generated by SYPROS is enriched with buggy goals and plans.

Diagnosing a Student's Solution

The process of diagnosing a student's program consists of three steps: To each program statement a *matching algorithm* finds all plans which match this statement. The *interpretation algorithm* reduces this one-to-many relation to an one-to-one relation, i.e. for each program statement it determines exactly one plan which is fulfilled by the statement. Finally, the *diagnosis algorithm* assigns one or more diagnosis categories like correct, incorrect, necessary, inconsistent etc. to each goal (cf. [HeGo93]).

The Interpretation Algorithm

The interpretation algorithm of SYPROS uses information about the student's knowledge and behavior which can be obtained from the student module. But most of this information is uncertain knowledge.

In order to investigate which model fits best this kind of knowledge we followed Pearl's classification of theories of uncertainty (cf. [Pear88]) and looked into the *neo-probabilistic* approach, e.g. Bayesian networks, into the *logicist* approach, e.g. default logic or truth maintenance systems, and into the *neo calculist* approach, e.g. Dempster Shafer-theory or fuzzy sets (cf. [Zade65]).

Comparing these models we found out that fuzzy techniques are well suited for the interpretation algorithm. Therefore, in SYPROS the uncertain information is modelled by fuzzy sets, and the interpretation component is implemented as an expert system based on fuzzy techniques (cf. [Zier93]).

Acknowledgement: This work is partly sponsored by the German Research Community (DFG).

References

[HeGo93] Herzog, C.; Gonschorek, M.: Generation of Adaptive Explanations in ITS with Hierarchical Knowledge Domains. In *Proc. International Conference on Computers in Education*, Taiwan, December 1993: pages 1–6.

[Pear88] Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1988

[Zade65] Zadeh, L.A.: Fuzzy Sets. In: *Information and Control*, **8**, 1965: pages 338–353.

[Zier93] Zierl, H.: Entwicklung und Implementierung von Heuristiken für die Interpretation und Diagnose von Studentenlösungen in einem intelligenten Lehrsystem. Diploma thesis, Institut für Informatik, TU München, 1993

666

Educational Multimedia Software Development Model: A Revised Empirical Approach

ENRIQUE HINOSTROZA, PEDRO HEPP
Enlaces Project
Universidad de La Frontera, P.O. Box 380, Temuco, Chile

Until now, traditional software engineering development models (DeGrace and Hulet Stahl, 1992) have failed to meet the particular requirements for multimedia software production in general (Brooks, 1987) and educational multimedia software in particular. There is an important need for a simple software development model for producing educational material, particularly, in countries with different cultural realities and where teachers are not technology aware. This paper is the first step to produce a set of guidelines that help them to go with the technology.

Our educational multimedia software development model highlights the following principles: (a) Early study and definition of the user, assessment of the pedagogical value and usability of the proposed software (Dockterman, 1991). (b) Easy to build and low cost combination of sketches and prototypes that help during the design and implementation. (c) Early and reiterative evaluation of the project in all of its dimensions (content, functionality, structure and usability). (d) Parallel work of the members of the team, allowing the evolution, evaluation and implementation (Boehm, 1989; Nielsen, 1992) of each component of the software separately but also integrating them into a tangible product at early stages.

We divide the software development model in five stages: (i) **Project Definition**: These are mainly brainstorming sessions of the working group (teacher, psychologist, engineer and art designer). Some of the points addressed here are: theme and contents, availability of information (text books, videos, etc.), the user and his/her environment and resources for development and use. (ii) **Project Design**: It has three main goals: to collect the maximal amount of information, to define the application's general structure and to formalize the design with a graphical representation of the components (like a softecture in Gilb 1988), the browsing structure and the helps available. This stage is divided in three minor steps that compound an iterative process with incremental contents and expressiveness. Each one represents an evaluation point, where the project is critically analyzed. (iii) **Application Evolution**. It is an iterative process in which users probe (test) alternates with rearranged computer prototypes. (iv) **Software Product Production**: This step includes all the additional and complementary material that the software should include in order to become a real market product. (v) **Maintenance**: It is not included as a real stage of the software development model, but must be considered in the development model because of the resources that it will demand to do it.

These five steps summarize the proposed software development model. Some additional considerations such as the role of each member of the team (designer, educator and engineer), the representation tools and the evaluation procedures, as well as the general features required for the end-product are currently being worked on.

Acknowledgments

This work is being supported by Fondecyt, with the research projects number 1930611 and number 1930569 and by the Chilean Ministry of Education through the MECE program in the project Enlaces (Hepp et al, 1993).

References

Ambron, S. & Hooper, K. (Eds.) (1990). *Learning with Interactive Multimedia: Developing and Using Multimedia Tools in Education*. Microsoft Press.

Boehm, B. W. (1988). A Spiral Model of Software Development and Enhancement. *IEEE Computer*, May, 61-72.

Brooks, F.P. Jr. (1987). No Silver Bullet: Essence and Accidents of Software Engineering. *IEEE Computer*, April.

Dockterman, D. A. (Ed.) 1991. *Great Teaching in the One Computer Classroom*. Tom Snyder Productions.

Gilb, T. 1988. *Principles of Software Engineering Management*. Addison Wesley.

Hepp P., Alvarez M. I., Hinostroza E., Laval E. 1993. 'La Plaza' A Software Design for an Educational Network *Proceedings of ED-MEDIA 93*, 241-246

An Environmental CAI-System In The Field Of Psychology

The Construction of CTP System
And
The Effectiveness of Classroom Demonstration Using CTP System

R. IHARA, A. KAWAMURA, AND C. KATOH
Department of Information Science
Sakuyo Junior College,
Okayama. J703 JAPAN

H. HAYASHI AND S. SAKATA
Department of Behavioral Sciences
Hiroshima University.
Higashi-Hiroshima. J724 JAPAN

S. ISHIHARA
Department of Management Inforamtion
Onomichi Junior College,
Onomichi. J722 JAPAN

K. HATAMOTO
Human Factors Laboratory
Hiroshima Chuo Women's Junior College,
Hiroshima. J731-01 JAPAN

A. YAMAGAMI
Department of Psychology
Konan Women's University.
Kobe. J658 JAPAN

Our "Computerized Textbook of Psychology"("CTP" for short) project has been promoting the development of a courseware for psychology on personal computers. Some parts of this study have been published elsewhere[1]. The aim of the CTP project is to develop an environmental CAI system so that it will provide a new technical environment for learning and teaching psychology, for not only students but also those scholors, who intend to reorganize the vast field of psychology. In contrast with conventional ICAI/ITS approach, which can not be applicable to such a field with large and/or non-linear knowledge structure as psychology, the CTP system brings the "browser frame" into hypertext structure, which help browsing and searching the target easily out of large knowledge space. The CTP system stands open to the selective modules in knowledge space in the forms of explanations, figures, animations and simulations.

Classroom Demonstrations Using CTP System

As a unique feature of CTP in contrast to other CAI systems, all the demonstration programs in the CTP system can be used for classroom demontration as well as self-educational system. We have conducted two experiments(120 subjects each) to examine the effectiveness of the classroom demonstration function of CTP. The purpose of the experiments was to assess the visibility and understandability of the classroom demonstration under regular classroom circumstances equipped with inexpensive audio-visual system. Although no explanation about the demonstrations was given to the subjects in the experiments, high visibility and understandability in such a way as the experimental results showed the demonstrations attracted the students' interest and made them experience psychological facts.

References

[1] R. Ihara, *et. al.*(CTP project): Developing A "Computerized Textbook Of Psychology".
IPSJ SIG Notes, Vol. 92, No. 77, pp. 19 27(1992), Vol.94, No.10 pp.1 10(1994) (in Japanese).
K. Hatamoto, *et. al.*(CTP project): The CTP project: computerized textbook of psychology.
Human-Computer Interaction: Applications and Case Studies, Elsevier, (1993) pp.444 449.

A Hypermedia-driven Computer Assisted Mathematics Remediation Package

J. James
Department of Applied Computing and Mathematics
University of Tasmania at Launceston, Tasmania, 7250, Australia

M.Taplin
Department of Education
University of Tasmania at Launceston, Tasmania, 7250, Australia

There has been a long-time concern about the adequacy of mathematical skills of many students entering primary and early childhood education courses. Because of resource implications, it has not been feasible to offer a remediation program. Therefore, it was undertaken to develop a computer assisted math remediation package.

Description of Requirements for the Package

Because there is a wide range of students' abilities and backgrounds, an underlying pre-requisite for the package is that it be self-paced. Furthermore, it was intended that the software developed be capable of catering for its users' different learning styles.

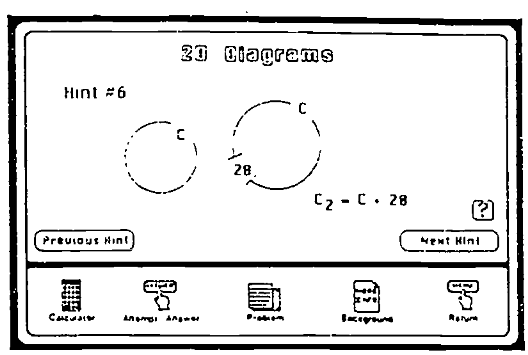
Analyses indicated that for this particular target group, the content needed to introduce students to unfamiliar mathematical content, particularly algebraic expressions and in doing this, equip them with skills to transfer knowledge to unfamiliar situations. Therefore, the package is intended to use problems which require a range of background mathematics knowledge and also require many steps in each problem's solution.

A group of 25 students and mathematicians was asked to solve a selection of problems and to describe the problem solving processes used. These processes were categorised as: visualising a 3-dimensional model of the problem; drawing a 2-dimensional representation; using equations, and using English expressions. For each problem, the package will offer a series of appropriate strategies across these four sub-headings.

Implementation of the Package in a Hypermedia Environment

The package is being developed using HyperCard to create a hypermedia environment. This will allow the incorporation of text and graphics into a user-friendly interface. The non-sequential nature of multiple problem solution paths is readily implementable using hypermedia. Marsh and Kumar (1992, p25) state that the multi-dimensional nature of hypermedia presentation is consistent with learning models of recent cognitive theories.

The figure at right gives a good indication of the user interface design. It can be seen that the student has access to a calculator; can leave the tutorial temporarily to attempt an answer; can have the original problem displayed at any time; has access to background knowledge, and can leave the present tutorial and return to the problem solving menu.



The advantages of developing a hypermedia system are consistent with Johnson and Grover's hypertutor model (1993, p7). These include learner control using an intuitive, consistent interface, immediate feedback, and easily accessible learning resources.

note: a full version of this paper is available from the authors.

References

Johnson, C., and Grover, P., (1993). "Hypertutor Therapy for Interactive Instruction", *Educational Technology*, Jan., 5-15.

Marsh, E., and Kumar, D., (1992). "Hypermedia: A Conceptual Framework for Science Education and Review of Recent Findings", *Journal of Educational Multimedia and Hypermedia*, v1, 25-37.

CASPER: A Hypermedia Departmental Information System

A. JOHNSON, F. FOTOUHI, N. GOEL, R. WEINAND, AND J. LECHVAR
Department of Computer Science
Wayne State University, Detroit, Michigan, 48202, U.S.A.

It is important for college students to have as much information available on their department as possible. This information should be up to date, and easily accessible and digestible. When students have a question, they should have the means to find their own answers quickly and easily.

To satisfy this need in our department we designed and implemented a hypermedia information system called CASPER - "Computer Assisted Studies Planning and Educational Resources." The CASPER software includes an Oracle database, a full colour Hypercard interface, and Quicktime movies. The CASPER hardware includes a Macintosh IIci with a 13" colour monitor, a mouse, and a laserprinter. The student interacts with CASPER using only the mouse.

CASPER contains information about our department, its instructors, its courses, and equivalent courses at other universities.

CASPER helps the student find information about our department. A student can see a list of presentations and other upcoming events, or read a list of answers to commonly asked questions. The student can see a full colour map of the department and click on rooms to see who it's occupants are.

CASPER helps the student find information on a particular instructor. A student can click on an instructor based on that faculty member's name, or their face from a set of full colour pictures. Clicking on a faculty member brings up information on that instructor including their name, title, phone number, office number, lab number, office hours, research interests, and teaching assignments for the current and future terms. The student can also play short full colour movies of the faculty members introducing themselves, allowing the student to see and hear them.

CASPER helps the student find information on a particular course. A student can click on a course from a list of all the courses taught in this department, or a list of all the courses taught this term. The student can see when and where the course is being taught, a sample syllabus for the course, and who the instructors will be for the next several semesters.

CASPER helps the student find informaton on equivalent courses at other universities. A student can click on a university and see which courses transfer to our department, or click on a course in our department and see a list of equivalent courses at other universities. This includes information on the equivalent course from the other university's Course Bulletin.

With all of this information available, CASPER helps the student choose which courses they will take in the coming terms. A student enters information about the courses they have previously taken, and the grades they have received. CASPER shows which courses remain to be scheduled. when those courses will be offered, and who will be teaching them. The student can then move term by term into the future scheduling courses. Once the student has created their plan of work they can print it out or save it onto a floppy disc. With the floppy disc, the student can return to CASPER at a future date and make changes to their plan.

CASPER has a friendly interface that makes the system fun to use, as well as informative - encouraging students to explore. CASPER allows the student to quickly and easily see related information. Clicking on an instructor's name or picture anywhere in the the system brings up information on that instructor. Clicking on a term brings up a complete list of courses offered that term. Clicking on a course brings up information on that course. Clicking on a room number brings up the departmental map showing where that room is located.

In February 1993, CASPER was placed in our department's main office where it is always left running. Student reaction has been overwhelmingly positive. They find the system very easy to use the first time they sit down in front of it. Their only complaints have been that the system should contain even more information.

CD-I: a Multi-media Tool for Distance Learners

Tom Jones
Department of Educational Psychology
University of Calgary, 2500 University Drive, Calgary, AB T2N 1N4, Canada

Ervin Schieman
Department of Curriculum and Instruction
University of Calgary, 2500 University Drive, Calgary, AB T2N 1N4, Canada

A relatively recent development in multimedia appears to be an excellent candidate as a wide-ranging resource for distance learners. CD-I (compact disc interactive) is both a media and a system specification and defines what can be present on a disc, how it is coded and organized and how disc/system compatibility can be maintained. Other advantages of CD-I include compatibility with the CD-Digital Audio specifications (the Red Book), compatibility with existing consumer electronic products (stereo systems and televisions), compatibility with other broadcast conventions (PAL, SECAM) and expansion possibilities which turn the CD-I system into a complete microcomputer system.

Given that interactivity has been defined in various ways in distance education and that interactivity is at the heart of CD-I, Lundin (1989) has identified six levels of interaction in the distance delivery of instruction. For this discussion, levels 3 through 5 are relevant:

- Level 3: 'limited interaction' in which the participant has choices regarding the exploration of the CD-I-based course content;
- Level 4: 'responses' which are requested from the learner;
- Level 5: 'simulated' interaction in which the material acts as a catalyst for real-time interaction among participants.

In each of these levels, the learner is in firm control of the pace of interaction and is actively engaged in the acquisition and mastery of the course-related skills and concepts. When using CD-I in level 3, the learner can choose to view the course materials from various perspectives if the disc designer has provided for unrestricted access to the disc's contents in addition to the navigational tools (screen design and interface). Level 4 interaction would be commonplace as CD-I is designed for exactly this. Level 5 would occur during the real-time videoconferencing and would allow for the use of a common source of information for all participants.

Propositions about the use of CD-I in distance education, related to the above three levels, appear to emerge:

- Proposition 1. CD-I technology can provide the distance learner with more opportunities to employ personalized learning strategies than traditional distance delivery systems,
- Proposition 2. CD-I technology has the potential to enhance and to facilitate the exploration and organization of new information;
- Proposition 3. CD-I technology has the capacity to reduce the obstacles inherent in technological interaction.

This new technology (new, at least, in the way in which it is employed in instruction) can offer increasingly more convenient and more effective channels of educational opportunities for CD-I is a system that is at once flexible, individualistic, comprehensive, and easy-to-use.

References

Lundin, R. (1989). Interaction and participation: The key elements in the design of educational videoconferencing. Paper delivered at the Pacific Telecommunications Eleventh Annual Conference, Honolulu, Hawaii.

Information Gathering Styles in an Interactive Environment: the Influence of Computer Literacy and Technology Use

DAVID KAMERER AND BARTON K. WILCOX
Elliott School of Communication, Wichita State University
Wichita, KS, USA 67260-0031

Multimedia software permits users to access information in many new ways compared to traditional media such as books. In this study, subjects were instructed to find information on aviation heroes of the 1920s in a multimedia encyclopedia (Compton's Interactive Encyclopedia on the CD-I platform). Our goal was to discover the relationship between information gathering styles and levels of computer literacy and technology use. We also created profiles of the most and least efficient searchers.

Subjects with high levels of computer literacy on three dimensions ("anxiety," "confidence," and "liking," as operationalized by Loyd and Gressard, 1986) were more efficient searchers than those with low computer literacy scores. Heavy technology users on two dimensions ("everyday technology" and "information seeking") were more efficient searchers than light technology users. However, the computer literacy and technology variables did not effectively predict satisfaction with the encyclopedia, which was high even among inefficient searchers (Kamerer and Wilcox, 1993).

The most efficient searchers typically used the "Title Finder," which is analogous to an alphabetical look-up. Title Finder searches were approximately twice as efficient as other, more associative approaches such as "Time Machine" (search by date) or "Fact Index" (search by category). The most efficient searchers were also more than twice as likely as inefficient searchers to use "hot key jumps," while the least efficient searchers were more likely to use a wide range of features. Inefficient searchers also spent a fair amount of time getting lost. In fact, four of 82 subjects failed to find the topic at all in 20 minutes.

In short, efficient searchers successfully used a few features, while inefficient searchers either explored many features or got bogged down in features which were inappropriate for the search. This suggests some implications for design of interactive materials. First, "bells and whistles" such as a "time machine," may help sell multimedia software, but they may not be very useful. Also, too many features seemed to create a kind of anomie in our subjects, who then were distracted by them at the cost of finding information. Content analysis of the sessions reveals that many of the iconic conventions of the interface are unclear to the user. The feature that seemed to most help searchers is the hypertext link, in which a user can jump from one article to another, related, article, without having to navigate back through earlier menus.

References

Kamerer, D. and Wilcox, B.K. (1993). *Computer Literacy, Technology Use and Compact Disc-Interactive: An Exploratory Study.* Presented at the annual meeting of the Association for Education in Journalism and Mass Communication, Kansas City, August.

Loyd, B. and Gressard, C. P. (1986). "Gender and Amount of Computer Experience of Teachers in Staff Development Programs: Effects on Computer Attitudes and Perceptions of the Usefulness of Computers." *AEDS Journal*, Summer, pp. 302-311.

Structured Authoring in the Hyper-PC Environment

F. KAPPE, N. SCHERBAKOV
Institute for Information Processing and Computer Supported New Media, Graz University of Technology, Schieszstattgasse 4a, A-8010 Austria.

V. LEBEDEV, A. NEDUMOV
St. Petersburg Technical University, Politechnicheskay 29, St. Petersburg, 195251 Russia

By looking at the current state of Computer Based Learning we see that we definitely deal with a new type of computer related product which is called *courseware*. Although there exist sophisticated and well-established methods of hardware and software design, there do not exist such methods for courseware preparation. In other words, in courseware design we face a situation similar to the very early stages of software design when authors developed their own isolated programs without taking into consideration reusability and modularity of the software developed. In analogy a new technique of courseware design has to be implemented in up-to-date authoring systems.

Hyper-PC is a comprehensive software tool running on PCs under MS-Windows for developing interactive multimedia courseware. It brings together text, graphics, animation, sound, images and answer-judging procedures in order to create computer-based training courses. It should be especially noted that the creation of all components of courseware is a programming-free process. Even quite sophisticated animation and answer-judging algorithms can be defined by means of a special non-procedural model of courseware being developed.

The frame-channel model is a paradigm which allows to formally define the structure of a large courseware product and thus manage the process of its development. Within this model, the internal structure of a courseware system is perceived as a *frame structure* which includes: a number of so-called *frames* and a number of *channels*, which are functional relationships ("links") between frames. A certain frame can be defined in the form of either a primitive node or a frame structure. Note the recursive definition which allows to apply the same model on different levels of abstraction. In analogy, channels can be seen as an unified approach to the interface between functional parts of a courseware product.

The main action which can be applied to a frame is to activate it. When a certain frame is activated, its body is evaluated. If the body is a primitive node, then this chunk of educational information is visualised on the screen. If the body is another frame structure, then the activation procedure is recursively applied to this structure. Channels serve as a predefined message passing mechanism.

Three levels of detail exist during the definition of a certain courseware object. On the first level, the prototype developer deals with a number of basic functions, and with the rules of frame type definition. On the second level, the prototype developers deal only with previously defined types of frames, i.e. with the current library of types. They can apply simple rules in order to build new frame types on the basis of the current library of types. On the third level, the user has got a number of frame types which can be interpreted as complete courseware objects. The subset of such courseware objects is a user's own courseware design philosophy.

There are some properties of our model which are of potential benefit from the point of view of the management of courseware projects:

- the model includes a clear and convenient graphic notation and can be easily metaphorised;
- the model can be applied on the different levels of courseware specification and implementation;
- the model supports rapid *prototyping*, including the possibility to apply a previous version of the courseware system within the latest version.

On Adventure Gaming as an Interface of an Educational Microworld

M.I.KARYAKIN, V.A.EREMEYEV, O.G.PUSTOVALOVA
Mechanics and Mathematics Department, Rostov State University,
5, Zorge str., Rostov-on-Don, 344104, Russia

One of the main form of teaching natural sciences is an experiment, practical and laboratory researches. But our experience tells that a participation in the real experiment and even successful completion of it does not mean successful acquirement of a knowledge minimum which is assumed to be obligatory. The main goal of the system of a computer support of an educational experiment on mechanics "Labquest" described below is to induce students to apply all required knowledge and, in that way, to ensure its acquirement.

Learning and gaming

The intercourse of the students with the system is realized in the form of a computer game. There exists a number of reasons which justify this choice.

Both at teaching and at the knowledge control a student must feel himself free and liberated, and the game gives an opportunity to eliminate superfluous tension and nervousness. The game-like form of the system leads to a stimulation of an interest and to substantial increase of motivation. The process of joint computer gaming assists to the students in closer unity of their working group. From various types of existing computer games we have chosen so called adventure games. The characteristic feature of these games is almost full freedom of game's hero to choose ways to his targets. This feature is very attractive for the creation of educational "world" someway adequate to real.

Educational and gaming problems

During the game a student has to solve educational problems of different types.

(i) Construction of a sentence. The student makes up sentences, formulae or diagrams by means of a given set of words, mathematical expressions and picture elements.

(ii) Calculation by formula. The student simply has to enter a number (e.g., value of critical load) which must be found beforehand by means of one of the main theoretical formulae.

(iii) Measurements. The aim of the player is to superpose screen images of the object and measuring tool, make necessary adjusting and read and interpret the readings.

Concrete representation of each problem may significantly changes even within the one type. It seems to be very important to coordinate with the plot an origin of every problem and necessity of its solution for further advancement.

When developing a game-like system it is not easy to resist the temptation to use some elements of traditional arcade games. But, as mentioned by G.Jones (1991), the criteria for success or failure in the simulation must be determined not by good hand-eye coordination but by the learning objectives. In the "Labquest" system arcade game sequences give an opportunity to cause the student to repeat some set of educational exercises and so to consolidate acquired habits and learned facts.

During the game the system constantly keeps a score. Such scoring allows a teacher to make prompt analysis of current situation, to determine the working stage at which students are situated. Moreover the increase of the score is an additional stimulus to the students.

References

Jones G (1991) Some Principles of Simulation Design in Interactive Video for Language Instruction.*Simulation & Gaming, 22*, 239-247.

Teachers as Developers of Multimedia Learning Environments: a Vehicle for Professional Development

PETER KENDAL AND ROD NASON
Centre for Mathematics and Science Education (CMSE)
Queensland University of Technology, Brisbane, Australia, 4059.

Multimedia systems can be classified under the categories identified by Joyce (1988) as either 'exploratory' or 'constructive'. In the former category, users interact with a previously developed multimedia system, often by merely browsing the information. In the latter, users are involved more actively with the multimedia system either using it to construct knowledge while completing some specific task or to participate in the development of the multimedia system itself. A number of authors (Hutchings, et al.,1992; Kass and McGee, 1993) propose that significant learning only occurs when users are involved in 'constructive' multimedia experiences.

This proposition underpins the approach adopted in this study involving nineteen practising teachers enrolled in a post-graduate course at the Queensland University of Technology. The aim of the study was to investigate how requiring teachers to develop and evaluate multimedia learning environments affects their understandings of, and confidence in, using multimedia systems.

Procedure

This study involved three stages: (1) Evaluating teachers' initial conceptions of multimedia by analysing the data collected from a free-write exercise using grounded theory techniques (Strauss & Corben, 1990). Teachers also completed a *Survey of Concerns (SoC) Questionnaire* (Hall et al., 1977) which provided a quick-scoring measure of stages of concern associated with their use of multimedia as part of their teaching. (2) Training teachers in multimedia materials development by requiring them to conceptualise, develop, and implement a multimedia-based learning environment. (3) Repeating the free-write and SoC activities and their analyses on completion of the course.

Main Findings

F1: On completion of the course, conceptions of multimedia focussed less on the technology involved and more on educational and people-related characteristics.

F2: Initial non-specific conceptions of interactivity were replaced by specific statements defining interactivity as a means by which users take control of what is learnt and how it is learnt.

F3: The SoC questionnaire responses indicated that the course of study was a useful form of professional development in satisfying teachers' initial concerns relating to awareness of and information on educational applications of interactive multimedia.

References

Hall, G., George, A. & Rutherford, W. (1977). *Measuring stages of concern about the innovation: A manual for the use of the SoC Questionnaire.* Austin, Texas: The University of Texas.

Hutchings, G., Hall, W., Briggs, J., Hammond, N., Kibby, M., McKnight, C., & Riley, D. (1992). Authoring and evaluation of Hypermedia for education. *Computer Education*, 18, 171-177.

Joyce, M.(1988). Siren shapes: exploratory and constructive hypertexts. *Academic Computing*, 10-14, 37-42.

Kass, A. & McGee, S. (1993). *Road Trip: Learning geography through simulated travel.* Paper presented at AERA.

Strauss, A. & Corbin, J. (1990) *Basics of qualitative research.* Newbury Park: Sage Publications.

A methodology for configuring simulation-based learning environments

MICHIEL KUYPER, ANJA VAN DER HULST, ROBERT DE HOOG and BERT BREDEWEG
University of Amsterdam, Social Science Informatics, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands, Tel (+31)-20-5256733, Fax (+31)-20-5256896, Email: michiel@swi.psy.uva.nl

Whether building information systems in general, or simulation-based learning environments (SLEs) in specific, it is preferable to avoid writing low-level code. The SMISLE (System for Multimedia Integrated Simulation Learning Environments) (de Jong et al, 1993) workbench aims to offer the author predefined tailored building blocks for building SLEs. A building block can be seen as a generic template for filling in knowledge about the subject matter, instructional measures, learner assessment or the interface.

With these building blocks in hand the author is faced with basically three tasks. He has to (i) analyse the required knowledge, (ii) mold the knowledge into a conceptual structure that is compatible with the existing building blocks and (iii) configure the building blocks into a consistent structure. To guide and monitor this process of analysis, conceptualisation and configuration the author is offered a methodology (Kuyper et al, 1993).

As with every design process, the authoring process is seen as an iterative process. Interim and partial solutions are constructed during the development process which are not part of the final solution. Design, in attempting to satisfy an initial set of goals, always results in the discovery of new goals. A workbench therefore should cater for change. The SMISLE achieves this by means of the modularity of its building blocks.

In addition, the authoring process is inherently unpredictable due to the idiosyncratic attributes of each individual situation and designer. That is why it seems ill-advised to prescribe the process of design. Well-advised support to an author should be directed at solving the local problems he faces, while leaving him the freedom of choosing his own path of development. This methodological viewpoint will be operationalised in the workbench by three different tools.

Firstly, the workbench contains a process-based entrance to the development task. Here, a decomposition of authoring tasks is displayed. The author can relate his current task to the overall development process and can denote the status of development of the different tasks. Also, representations of tasks that are computer supported give access to this support.

Secondly, an agenda mechanism is devised to support the author in linking building blocks. This mechanism keeps track of all the open plugs and provides immediate access to them by name at any desired position in the SLEs architecture. In this way, the burden of switching between parts of the SLE while searching for open plugs is removed.

Finally, monitors in the workbench see to it that the author does not fill in wrong values in building block slots. Because authors are watched on this syntactical level, they are spared the tediousness of having to debug unnecessary errors.

References

de Jong,T., van Joolingen,W., de Hoog,R., Lapied,R., Scott,D. and Valent,R. (1993). SMISLE: System for Multimedia Integrated Simulation Learning Environments. *Paper presented at the DELTA workshop Design and production of multimedia and simulation-based learning material*, Barcelona, April 26-27.

Kuyper,M., Bredeweg,B., de Hoog,R. and van der Hulst,A. (1993). Authoring Methodology. Technical report University of Amsterdam DELTA/SMISLE/D16.

Acknowledgements

The work presented here was partially carried out under the DELTA programme of the EC as project D2007. The partners in this project are University of Twente (The Netherlands), University of Amsterdam (The Netherlands), Marconi Simulation (Scotland), Engineering Systems International (France) and Framentec (France).

A computer-managed learning environment for students of anatomy

Harry B. Lee and Ross Kemp,
School of Physiotherapy,
Curtin University of Technology, Shenton Park, Western Australia 6008.

Research has shown a Computer Managed Learning Environment (CMLE) to be most effective when learners are empowered to access materials to meet their own needs (Lee & Allison, 1992 & 1993; Lee & Cameron, 1994). Rates of progress can be accelerated by the ability to monitor individual levels of competence by instant feed-back (Lee, 1994). In a project funded by the Committee for the Advancement of University Teaching (CAUT) in Australia, a Self Operated Computer Controlled Educational Resource (SOCCER) was developed to provide a cost-efficient navigational instrument to Interactive Computer-assisted Learning Programs (ICALP) to enable physiotherapy students to assess their own performance whilst learning anatomy (Lee, 1994). Using SOCCER, each individual is registered by password with unlimited access to learning and testing materials. All activities are continuously tracked and recorded for evaluation thereafter. These data acquire information about learning from pre-tests and the acquisition of new information during progression to post-tests under controlled conditions for continuous review by learners and administrators alike. In this way SOCCER provides access to ICALP's, interactive Multiple Choice Question Test Items and interactive True-False Test Items in a sensitive learning environment with a Feed-Back Instrument to evaluate learner reactions to the CMLE.

Learning anatomy and validation of SOCCER by comparative studies

Jones, Olafson and Sutin (1978) compared freshmen medical students with those in traditional classes to show that they could learn gross anatomy equally well without lectures or dissection. Walsh and Bohn (1990) used CAL to teach gross anatomy to 48 medical students from a class of 151, to find no significant difference between the two groups. In the same way, Lee & Allison (1992) could find no differences between randomly selected groups of physiotherapy students whilst learning anatomy of the lower limbs, either by lecture or by ICALP.

In a further comparative study, Lee & Allison (1993) converted identical neuroanatomical information from ICALP's into lectures for computerised overhead projection to 41 students by lectures and to 37 students by ICALP. At pre-test, the mean for each group was 43.58 and 44.81, with a standard deviation of 8.67 and 6.54 respectively. At post-test, the lecture group achieved a mean of 73.19, with a 7.76 SD, whilst the ICALP group achieved a mean of 75.42, with a SD of 6.76. The same test was unexpectedly re-applied at 60 and 120 day intervals. The lecture group obtained 68.04, with a SD of 8.49 after 60 days, and 68.06 and 8.24 after 120 days. Whilst the ICALP group achieved 70.34, with a SD of 9.58 after 60 days, and 69.68 and 9.02 after 120 days. These results were validated by replication studies in neuro and cardio-pulmonary anatomy to subsequent student groups (Lee, 1994). From this, it is clear that SOCCER in a CMLE can facilitate ICALP software to provide individual learners with a continual review of the rate and quality of their progress together with instant feedback at the end of each procedure. These methods provide a continuous source of research data to empower learners to take responsibility for their own rate of progress, setting aside the need for formal lectures in anatomy in a climate of problem-based learning and thinking at a more advanced level than could otherwise be expected.

References

Jones, N.A., Olafson, R.P. & Sutin, J.S. (1978) Evaluation of a gross anatomy program without dissection. *J. of Medical Education,* March 53, 3, 198-205.

Lee, H.B. & Allison, G. (1992) A comparative study of the presentation of anatomy by lectures versus ICAL packages to physiotherapy students. In: Proceedings of the *International Interactive Multimedia Symposium,* Perth, W.A., Jan 27 - 31, 235 - 245.

Lee, H.B. & Allison, G. (1993) Learning anatomy at the tertiary level by computer. In: Proceedings *Forum* 1993. In Press.

Lee, H.B. & Cameron, D. (1994) Strategies to provide pathways for learners in a computer-managed learning environment. In: Proceedings of the *International Interactive Multimedia Symposium,* Perth, W.A., Jan 23-28, 273 - 283.

Lee, H.B. (1994) A comparison of the cost-effectiveness of traditional lectures with interactive computer-assisted learning packages. In: Proceedings of the *International Interactive Multimedia Symposium,* Perth, W.A., Jan 23-28, 264 - 272.

Walsh, R.J. & Bohn, R.C. (1990) Computer-assisted instructions : a role in teaching human gross anatomy. *Medical Education* November, 24, 6, 499-506.

Continuing distance education tools: Relevance of a contextual evaluation

J. LÉGARÉ AND S. DOUZOU
ITOT Program
Centre for Information Technology Innovation, 1575 Chomedey, Laval, Québec, H7V 2X2, Canada

Adult education and training are increasingly relying on telecommunication and information technologies to support continuing distance education, on-the-job training and even on-the-job performance. Models of adult education suggest that the learner's motivation to learn is not simply instrumental and situational but in a large part developmental (Berbaum 1984; Piaget *in* Inhelder *et al.*, 1977). Adult training innovations have to take this into account and go beyond mere instrumental support that can take the form of a "crutch" instead of an educational tool. These on-the-job educational/training tools are highly context dependant, whether for their design, their use, or their impacts. Questions have arisen as to whether the sophisticated technologies designed to support performance can contribute to developmental as well as instrumental educational goals, how, to what extent and under what conditions. We addressed these questions by conducting a contextual evaluation of the implementation of a continuing medical education technological tool for on-the-job performance support during a 14 month pilot project. The tool was a networked based information system aimed at developing competence and interest of general practitioners in the diagnosis and treatment of mental health problems through support of their day-to-day practice. The system had three components: 1) an expert consulting network using electronic mail; 2) an electronic education bulletin in the mental health field; and 3) a clinical pharmacology databank, which was a complementary incentive service. The implicit educational objective was to induce a shift in physicians' patterns of practice (a developmental goal, not a mere instrumental objective), away from a dependance upon prescription of drugs and towards adoption of alternative approaches through easy distance access to specialists (psychiatrists, psychologists, legal experts, etc.). As mental health problems are not exclusively treated by general practitioners, 62 professionals in 9 different types of healthcare settings participated in our case-study type evaluation: 25 physicians, 16 social workers, 14 nurses and 7 psychologists. We adopted an interpretive approach relying on the institutional aspects and interpretive flexibility of technologies (Kling & Iacono, 1989; Orlikowski, 1992) and on the search for the significances which different groups of actors involved in the process attribute to professional training and to the technological change (Jodelet, 1989).

In order for the expected changes in users' behavior or attitudes to occur as the result of using the training technology, the evaluation showed that it is important that the general practitioners themselves, and not just the experts or the promoters, consider the new pattern of practice supported by the technology as a best practice in the domain and as a possible practice in their organizational context. Otherwise, the users divert the technology from its intended purpose to reinforce their own current behavior. We found that confirmation of this diversion effect of the technology was related to the professionals' conception of training, to the constraints of their work context and to the stakes they invested in the tool. On-the-job intervention support was used to reinforce existing practice patterns of the physicians and did not encourage developmental learning and associated broadening of perspectives in treating mental health problems.

References

Berbaum, J. (1984). *Apprentissage et formation*, Que-sais-je?, P.U.F., Paris.

Jodelet, D.(ed.) (1989). *Les représentations sociales*. Paris : Sociologie d'aujourd'hui, PUF.

Kling, R. and Iacono, S.(1989). "The Institutional Character of Computerized Information Systems." *Office, Technology and People*. 5(1), 5-27.

Orlikowski, W.J.(1992)."The Duality of Technology: Rethinking the Concept of Technology in Organizations." *Organization Science*, 3(3), 398-427.

Piaget, J., in Inhelder, B., Garcia, R., Vonèche, J. (eds), (1977). *Épistémologie génétique et équilibration*, Paris, Delachaux et Niestlé.

Smart Instructor's Resources: Make Lectures Interactive Using Computer-Generated Projections

Dr. John S. Martin
Department of Chemistry, The University of Alberta, Edmonton, Alberta, Canada T6G 2G2

Performing the experiments that support even one of the many principles in a typical introductory course would, in most cases, take a lifetime. Attempts to set up a "discovery learning" laboratory usually result in the discovery of why it took thousands of very intelligent scientists several hundred years to work out all of the observations and principles that make up the content of a typical introductory University course.

One way to alleviate this problem, and put a significant amount of laboratory evidence into a course, is to use simulations. These allow one to put before a class experimental evidence that would normally be too expensive, time-consuming or dangerous. There are many excellent audiovisual presentations, in a great number of media, that allow a lecturer to bring into class realistic representations of critical experiments. The difficulty is that they generally are "canned" performances, complete in themselves, so that the lecturer must adapt his or her presentation to them, and follow their pedagogical strategy.

Now that many lecture halls have computer projection equipment, it is possible to take advantage of the intelligence and graphical capabilities of computers. I have designed interactive computer animations which simulate a laboratory, and provide unlimited and immediate access to any appropriate experimental apparatus. The lecturer has complete control over the on-screen events, so that they support rather than constrain his or her pedagogical approach.

These programs are called "Smart Instructor's Resources" or SIRs. They run on any modern DOS-based microcomputer with high resolution colour graphics (vga) and a mouse. SIRs use an intuitive mouse-based interface. This is critical in a classroom setting. An instructor is entirely occupied with instructional material and interaction with the class; it is unreasonable to expect her or him to puzzle over the operation of the computer as well. The interface is simple enough that the instructor can think of chemistry and instructional strategy, and be confident that whatever is needed can be called up onto the screen immediately.

The instructor and class together may explore a phenomenon as if they were in the laboratory, and thus retrace the actual discovery of physical and chemical principles. They may decide what to do next: in effect, ask "What would happen if...", and, if their question is properly framed (here instructors earn their pay!) immediately receive a meaningful answer from the simulation. After a suitable sequence of experiments, the class and instructor together may attempt to infer the underlying chemical principles. Students identify with and retain better scientific principles which in some measure they have found out themselves, rather than have had dictated to them by authority.

During the 1993-94 academic year I have used about twenty of these SIRs in my classes. They cover most of the topics of a standard introductory Chemistry course: reaction stoichiometry, atomic structure and the periodic table, behaviour of liquids and gases, calorimetry, acid-base equilibrium, redox and electrochemistry and chemical kinetics.

Students' reactions have been mainly enthusiastic, although some seem rather startled by being asked to switch to active learning in place of their customary passive note-taking role.

This work was begun while the author held a CATALYST Fellowship from IBM and the American Chemical Society at the University of Texas in Austin in 1992-93. The Smart Instructors' Resources have been submitted to the *Journal of Chemical Education: Software*, which makes available executable copies of software that it publishes.

A Prototype Digital Science Center

Miriam Masullo
T.J. Watson Research Center
Yorktown Heights, NY

Albert Snow
The Discovery Museum
Bridgeport, CT

By the end of this century large scale digital libraries containing the collective legacy of human knowledge and information will be accessed on-demand over gigabit networks. Books, videos and other traditional sources of information will be incorporated into these digital libraries in an almost straight forward move towards exploitation of the new technologies. That will not be so with other equally valuable sources of information, such as museums and science and technology centers. And, much research will be needed in order to understand how to effectively capture and represent learning experiences in digital libraries. But when we do this, equity access to such experiences will become a reality.

The intent of a science and technology center is to engage its young patrons in stimulating multi-sensory experiences in the hope of imparting knowledge and encouraging curiosity. The most obvious problem in doing that is access to the facilities. Thus, the educational value of museums and science and technology centers is currently minimally exploited. Teachers could make use of the educational value of the exhibits many times during the course of a single school year. But in fact students may visit a particular center only once during their entire K-12 years.

Capturing these experiences as interactive multimedia documents would facilitate integration of specific exhibits or groups of exhibits on related topics. into lesson plans. It would permit gathering related exhibits (in various center) for comparison and re-enforcement of ideas. But most of all, it would bring these experiences to students that might never otherwise have access to them. Using technology it will be possible to extend the cultural power of museums, and science and technology centers beyond their physical boundaries. It will be possible to reach students and teachers with a wealth of information, for repeated analysis at individual pace. And, it will be possible to access them when they would be most relevant and integrate them with curriculum.

But, bringing these experiences to a remote site using multimedia technology presents some significant problems. The most obvious problem is the lack of interaction with the physical objects contained in the museum itself. The challenge is to come up with ways to capture the spirit of the museum and convey it to the remote user. Another problem is that it requires a lot of planning and effort to integrate these experiences (in its existing physical form) into lesson plans and in general curriculum. Therefore, in general these experiences are sometimes regarded as close to casual educational activities, undertaken only as class trips and family outings, and viewed mostly as only motivational. The digital museum and science technology center have the potential of changing that, but it will require the development of curriculum models that help to integrate digital experiences with classroom activities.

The Discovery Museum and IBM's T.J. Watson Research Center have collaborated to explore these issues. Experimentally, various exhibits have been captured using full-motion video, audio, graphics and text to create multimedia documents representing many experiences in digital form. The digital experiences are interactive and stimulating, much like the physical ones. Work is still in progress, but early results suggest much promise for the approach.

The Study of Student Conceptions in Geological Mapping: A Phenomenographic Approach

J. MCCRACKEN, D. LAURILLARD
Institute of Educational Technology, Open University
Walton Hall, Milton Keynes, MK7 6AA, England

This study of student conceptions in interpreting geological maps was carried out following the tradition of a research methodology referred to as phenomenography which seeks to discover and describe the qualitatively different ways in which students think about a particular topic as they are in the process of learning. The general aim of phenomenography is to use samples of student conceptions to construct categories which are expected to have some generality in terms of the student population as a whole. (Francis, 1993). These categories of description, also referred to as an 'outcome space of conceptions' reflect a researchers' analysis of emerging patterns in the way students describe both their approach to a task or problem presented and the reasoning they apply to the problem.

There were two aims for this study. The first was to gain insight into the difficulties students experience in learning to interpret geological maps. The second was to consider how the results of this kind of study could be used to inform the instructional design decision-making process.

Conceptual Difficulties and Learning Approaches

The study of geological maps requires the ability to "infer the meanings of patterns found in rocks" (Chadwick, 1978) and the ability to visualize three-dimensional structures which have changed with respect to the dimension of time. Students in this study experienced three types of conceptual difficulties in learning to interpret maps:

1. Students confuse the relationship between present-day topography and sub-surface geological structures.
2. Students have difficulty in determining whether a sub-surface structure is an anticline or syncline.
3. Students have difficulty in visualizing three-dimensional sub-structures.

Students also displayed two types of approaches to learning to interpret maps. The first and most common approach, referred to as 'surface' (Entwistle & Marton, 1984) was characterised by the student attending to discrete features of the map. A 'deep' approach was characterised as the student first gaining an overall idea of what the map represented, followed by positing a theory of the history of the events which may have occurred, then looking for features on the map which supported the theory.

A number of implications for instructional design can be drawn from these results, including clarifying instructional goals, developing sequencing, developing graphical representations, and guiding the student in their approach to the task.

References

Chadwick, P. (1978). Some aspects of the development of geological thinking: *Geology Teaching*, v.2, 142-148
Entwistle,N.J. & Marton, F. (1984). Changing conceptions of learning and research, in F. Marton et al (eds) *The Experience of Learning*, Edinburgh, Scottish Academic Press
Laurillard, D. (1993). *Rethinking University Teaching*, London, Routledge
Francis, H. (1993). Advancing phenomenography: Questions of methods, *Nordisk Pedagogik*. V.2

Acknowledgements

This research is supported in part by the Alberta Research Council of Canada, and an Overseas Research Student Award by the British Department of Education and Science.

Student demographic composition in a multi-media class: Effects on attitudes towards computer-assisted teaching

ANANDA MITRA
Survey Research Laboratory
University of Illinois, Urbana-Champaign, IL 61801

This paper is based on the findings of an evaluation of the attitudes and opinions of students towards the use of multi-media in a plant biology course. The course selected for the case study was an introductory plant biology class that was a requirement for all university students. The course was taught in a large classroom to about 230 students. The lecture used a multi-media platform for projecting a variety of images on a large back-projection screen. The Focus-Question-Analyze-Recommend (FQAR) methodology was used to do the evaluation. The process consisted of five focus group discussions with randomly selected students from the class. The final questionnaire consisted of four pages with separate sections that addressed student attitude and opinions, perceptions of the effectiveness of the tools, perception of the competence of the instructors, suggestions for improving the course and student demographics.

It was found that the students had differential exposure to computer and computer use. In general it was found that most of the students either frequently or very frequently used word processing. The results also indicated that a private dormitory or apartment room was the place where computers were most often used. Moreover, about 53% of the students indicated that they had taken at least one course in the year prior to the study where computer-assistance was used in teaching. Based on such findings it is possible to categorize the audience of multi-media assisted courses into several different groups and then ascertain if indeed there is a difference between their attitudes towards the introduction of computers in teaching. The primary categories are: 1) students who have taken computer assisted courses compared those who have not, 2) students who have taken courses that require computer use as compared to those who have not 3) students who are regular users of computers outside of the classroom as compared to those who are not, 4) gender and 5) year in school.

Those who had been enrolled in computer assisted courses agreed that they felt comfortable using computers while the students who were not in such classes felt less agreeable towards the same issue. This suggests that other than attitudes towards personal use of computers, attitudes towards other computer-related issues are not particularly influenced by prior attendance in computer-assisted courses. This finding is supported by the fact that no significant differences were observed between the groups of students who were previously exposed to courses that had computer use as a requirement and students who had never taken such courses.

Noticeable differences were also observed between male and female students in terms of how the different genders felt about computer use and the introduction of computing in the work place. Women consistently had a negative image about the introduction of computers in teaching. This is a trend that needs to be observed carefully since this could suggest that instructors and course administrators need to pay more careful attention to the gender composition of their classes to see if indeed there is a need to modify the message or the computing techniques being used to teach the course. Finally, there were some significant differences between the ways the lower level students (Freshmen and Sophomore) and upper level students felt about computing in education and teaching. In general, the lower level students were less excited about the rapid introduction of computers in teaching and felt less comfortable using computers themselves. These findings highlight two specific issues. First, the classroom of the nineties is indeed a varied one and there are a large number of vectors of difference that cut across the classroom. This is important since the second finding of this study is that there are several instances where it is possible to find significant difference in attitudes of the students based on their demographic background. It is particularly important to know these differentials since this could ultimately impact the level of learning and the overall evaluation of a particular learning tool.

Towards a Hypermedia System for Teaching an Introduction to Computer Systems

Ahmad Nasri
Department of Mathematics
American University of Beirut, Beirut, Lebanon

Multimedia systems are receiving a great deal of attention. Hypermedia systems is a combination of such systems and hypertext. These systems have a great potential in education and in particular in computer science education. This paper reports on the developement of a hypermedia system for teaching an introductory course to computer system which will provide an ideal interactive environment for learning, teaching, designing and implementing students projects in related topics. The system consists of 10 nodes covering most of the traditional topics in the course. These are the History node, the Elementary Circuits node, the Digital Functions node, the Data node, the Organization node, the Peripherals node, the Simple Computer node and the Laboratory node. Finally, two more nodes, the Applications node and the IC Industry node, can be added to the system. A node may hold links to one or more sub-nodes each of which is designated for a sub-area of the topic involved.

MPW Object Pascal was used to develop several nodes such as the simple Computer. Other subnodes are built using Authorware Professional, Hypercard and MacApp. Ready packages such MacCAD, LogicWork and DigSim will be integrated into the design and analysis subnodes. Parts of the movies will be produced using existing software such as Macromind Director, Film maker and QuickTime. It is expected that the complete hypermedia prototype will be a living laboratory allowing students to navigate through the materials and to interactively modify the parameter of an environment in a particular node and study the results obtained.

Combining Hypermedia Authoring and Classroom Networking in One Software Environment

R. L. OAKMAN, J. A. WALLER, R. A. RISER, AND M. R. BLOEMEKE
Department of Computer Science,
University of South Carolina, Columbia, SC 29208, USA

Studies show that whereas people retain only about 10 percent of what they see and 20 percent of what they hear, they can remember about half of what they see and hear together and even 80 per cent of what they see, hear, and do. The educational advantages of multimedia presentation like television or movies (sight and sound) are obvious but even more so for interactive multimedia, with its component of interaction and response provided by the learner (Begley, 1993). Main advantages include the ability to share different sorts of resources on a computer, all stored in a similar technology, and user interactivity among different sources of information. A hypertext document is a wonderful multimedia tool, blending a variety of related materials into an interactive, highly visual mode of delivery. Readers can move around in a hypertext document at their own pace and direction, interacting with the material in a nonlinear fashion. But hypermedia applications are normally single-user and do not support collaboration and groupwork.

The educational software development group led by Dr. Robert Oakman focuses on multimedia authoring and groupwork editing tools. Since 1986 this laboratory has completed three nationally distributed pieces of educational software for the Macintosh. LiveWriter (Research Design Associates, 1990) provides interactive networked writing instruction. In LiveWriter a teacher can log onto a student's computer, see what he or she is writing, carry on dialogue about the work, and actually do editing simultaneously with the student. In 1991 LanguageWriter (Research Design Associates) added digitized voice recording and transmission features to the LiveWriter foundation. Teachers can record foreign language lessons in both text and voice; then students can read and listen to them and respond with either a written or voice response.

Expanding on the pedagogical ideas of the earlier programs, MediaLink (1993) allows teachers and students to author multimedia lessons (text, sound, graphics, digitized photographs, and movies) and then enter on-line collaboration for group work about these materials. Teachers and students can prepare their own Macintosh multimedia lessons by assembling a variety of text materials, sound files, digitized photos, or QuickTime movies. The program then becomes a collaborative learning environment with shared files and group work in a networked classroom. MediaLink allows sharing different sorts of resources on a computer, all stored in a similar technology, and user interactivity among different sources of information. The "drag and drop" interface used for linking resources and network users remains simple enough so that teachers and students can avoid the steeper learning curve of other hypermedia authoring tools and still be able to work together collaboratively. With MediaLink, the research paper of the future can be collaboratively created by several students and include a variety of multimedia materials.

Adding multimedia transmission and student feedback through the TCP/IP protocols of the Internet will make MediaLink a powerful distance learning tool. In fact, we are currently in the process of expanding these capabilities for the program to work in a long-distance environment. We are aiming for the same seamless delivery of multimedia materials and rapid user feedback through a broad band communication channel that we already have in a local area network. Research issues include the possibility of getting lost in cyberspace and the etiquette for turn-taking when multiple users are working cooperatively on the same MediaLink lesson.

References

Begley, Sharon. (1993) "Teaching Minds to Fly with Discs and Mice." *Newsweek* (May 31), 47.

Oakman, R. L., Waller, J. A., and Fenimore, F. (1993) "MediaLink: A New Method for Authoring Multimedia Lessons for the Classroom." *Proceedings of the Second Annual Conference of the Association for Applied Interactive Multimedia*, Savannah, GA, pp. 70-71.

A development of multimedia CAI software with the expert system for learning environment problems

TOSHIO OKAMOTO, NAGAHISA KAWASHIMA
The Graduate School of Information systems,
University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, 182, Japan

SADAO KATO
Uchida Yoko Co., Ltd.
2-6-16 Shinkawa, Chuo-ku, 104, Japan

Recently, the rapid technological advancement, the industrial activities, the sharp increases in population among developing countries, and the more mutually-dependent-international relations are altering many environments such as the one surrounding the earth.

In line with this trend, the necessity of preservation of the earth environment as a whole has begun to be recognized. How to make a balance between the environmental effects of human activities and preservation of the environment is a serious issue. To solve the environmental problems, the promotion of environmental education is necessary to educate the people to have good sensitivity and knowledge about the environment.

To reach these goals, we have developed a multimedia-based educational software which focuses on the environmental problem. This software has been developed for 5th and 6th grade students of a elementary school. In this study, cognitive science theory such as situated learning and knowledge-constructivism is considered. It is significant to provide the learning environment in which students can reconstruct their own knowledge through hypothesis-verification under the similar situated contexts.

The learning environment of this system is comprised of phase 1 and phase 2. Firstly, the students study the environment problems in the phase 1. Secondly, They go on the phase 2. In the phase 1, a student is requested to be a mayor who has the intelligence to improve the environment appropriately. The student can study the schemata and the causal relationship of environmental problems, which are classified and ordered systematically. In the phase 2, a simulator of designing a city is provided. A student becomes a mayor and is asked to make the city harmonized with nature under some constraints. The system illustrates the specified positions to become problem on the map, and gives the explanation of the consulting result by using the effective multimedia. The system shows the line of reasoning, and instructs the basic knowledge related to the situation of the region, with hypermedia of phase 1. In this manner, a student can learn the environment problems by himself. Moreover, the phase 2 is linked with the objects in the phase 1. Therefore, the system can give some advices which is suitable for each student utilizing the search method of objects-linkage in the phase 1. The teacher can arbitrarily set the system parameters by which the details-level of explanation is customized according to a educational purpose.

This system can teach a student the qualitative concept and superordinate knowledge of environmental problems. We have developed this system aiming to provide the learning environment under which students could raise up their critical mind and construct the profound knowledge of problem solving. Our further tasks are to develop the planning typed expert system in this frame work. Moreover, we need the function to have a student understood the scientific knowledge of environmental problems with qualitative simulators.

References

Fischer,G. & Reeves,B. (1992). Beyond Intelligent Interfaces : Exploring, Analyzing, and Creating Success Models of Cooperative Problem Solving. *Journal of Applied Intelligence 1*, 311-332.

Toshio Okamoto, Sadao Kato, Nagahisa Kawashima, and Tadashi Miyamae (1993). The intelligent interactive multi-media system for design-oriented problem solving: A function to explain the line of reasoning. *IEICE Tech. Rep., ET93* (105), 105-112.

A Multiple Knowledge Organization Environment
The Case of "The Emergence of the State of Israel"

A. OREN
Tel-Aviv University, School of Education , Knowledge Technology Lab, Ramat-Aviv, 69978, P.O.B. 39040, Israel

The project presented in this paper is based on the following theoretical assumptions:
1. Computers perform in learning environments as knowledge tools (Chen & Oren, 1991). This concept is manifested in the environment hereby discussed through the variety of information items and the inclusion of various tools needed for producing knowledge out of any information item.
2. People differ in the way they interact with Information systems. Factors like cognitive styles, search objectives, former experience with computers and the type of learning assignments might have an influence on the search strategy they use (Marchionini & Shneiderman, 1988; McNeer, 1991; Oren & Chen, 1992). The knowledge environment described supports these differences by classifying information in various ways, as well as create hypertext links.
3. Developing an autonomous learner is a main goal of an educational system. A learning environment, as the one we describe, allows the students to control the information search process and the knowledge production.
4. Developing historical thinking is a main goal of history instruction. Analysis of these objectives shows that learning by research and discovery is basically information handling like: searching for relevant information, analyzing different sources and evaluating relevance of information (Morris, 1991 ; Egartua, 1991).

The knowledge environment we developed has three main features enabling to realize it's learning potential:
A flexible knowledge organization, a curriculum orientation and a self-content structure.
* The flexible knowledge organization is achieved through various ways of information accessing: a menu-driven content list or Boolean queries, different modes of knowledge presentation or levels of information complexity, as well as browsing around in an hypertext mode.
* The curriculum orientation is manifested through the selection, amount and variance of information items which are important for comprehending the historical subjects: texts (written especially for the system, documents and historical research, memories and dairies, poetry and literature), graphical items (pictures, map, caricatures, schemes and diagrams) and voices (speeches and songs).
* The self-content structure of the system is manifested through a dictionary explaining difficult words, a lexicon supplying information about people, institutions and terms, atlas of maps, a synchronic time-line and authentic voices. A note book integrated enables the students, while collecting information, to write down their ideas or to copy extracts from information items.

References

Chen D. & Oren, A. (1991). Knowledge Technology and Curriculum Theory. In *International Encyclopedia of Educational Technology*. Pergamon press. 131-137.
Igartua, J.E., (1991). The Computer and the Historian's Work. *History and Computing*, 3 (2), 73-83.
McNeer, E.J. (1991). Learning Theories and Library Instruction. *The J. of Academic Librarianship*, 17 (5), 294-297
Marchionini, G., & Shneiderman, B. (1988). Finding Facts vs. Browsing Knowledge in Hypertext Systems. *IEEE Computer*, 21 (1), 70-80.
Morris, R.J. (1991). History and Computing: Expansion and Achievements. *Social Science Computer Review*, 9 (2), 215-230.
Oren, A., Chen, D. (1992). New Knowledge Organization in the History Classroom. *History and Computing*, 3 (2), 120-131.

Acknowledgments
The project described below was supported and developed in cooperation with the Curriculum Development Department - the Ministry of Education and Culture in Israel, and was applied through a system developed in the Center of Educational Technology (CET)

Remote Learning and Training Services: How Disabled People can use Multimedia Applications

NELSON PACHECO DA ROCHA
Universidade de Aveiro / INESC, Campo Universitário, 3800 Aveiro, Portugal

LEONOR MONIZ PEREIRA
Faculdade Motricidade Humana, Estrada da Costa - Cruz Quebrada, 1499 Lisboa, Portugal

CLARA CIDADE
TLP, Rua Passos Manuel - 2, 1100 Lisboa, Portugal

ERNESTO AFONSO, PEDRO BREDA
INESC, Campo Universitário, 3800 Aveiro, Portugal

The multimedia systems can be particularly suitable for the intervention for people with special needs. This intervention should comprise the following aspects: restoring the loss of functions, compensating for the disability by enhancing other skills, compensating for disability through the use of technical and not-technical aids and changing the environment to adapt it to the skills of the person.

The TeleCommunity project in the context of RACE II (Research and technology development in Advanced Communications technologies in Europe) is a set of Advanced Communications Experiments (ACEs) which aim to carry out work in advanced communications and is concerned with how the transition to Integrated Broadband Communication (IBC) affects the implementation of advanced services for people with special needs. Within TeleCommunity, the Portuguese ACE exploits the potential of ISDN in the support of disabled people, namely visual, mental impaired and elderly, both in the social centres' context and in the support of residential patient. The different provided services, namely Learning and Training services, allow the target users the access to further specialise support (Pereira L.M., Rocha, N.P., Cidade, C., Lebre, P., Purificação, J., 1993).

The terminal equipment for these remote care services are modular multimedia terminals, based in personal computers with video codecs (Tandberg Vision Model 15), and whose development is one of the goals of our participation. Each terminal offers an integrated capability to set-up, simultaneously, point-to-point videophony and data connections, typically between a service provider and a client.

For the user interface, both the service centre and client terminal user interface, a consistent metaphor has been established: the Virtual Resources Centre (Rocha, N., Cidade, C., 1993). This metaphor is based on the notion of a three dimensional room with different pieces of furniture and equipment, each one representing a different application (tool) with functions to establish connections over ISDN and with hypermedia facilities for preparing, presenting, accessing and discussing information. These hypermedia facilities obey to a server-client scheme that has been implemented with Object Linking and Embedding (OLE) technology, which able the use of commercial OLE servers.

During the pilot experience most of the users had demonstrated satisfaction (only 11% of the users had show inhibition, insecurity or tension behaviours), and they present good performance in the use of the facilities (Pereira L.M., Rocha, N.P., Cidade, C., Lebre, P., Purificação, J., 1993). Considering the staff opinion, they considered easy to use the equipment and when asked about the viability of the services they referred that the disabled users could benefit very much from these types of services. Furthermore, they considered essential the use of the videophony, otherwise they could not see the users that could cause a lack of "human contact".

References

Pereira, L.M., Rocha, N.P., Cidade, C., Lebre, P., Purificação, J. (1993). Visual, Mental Impaired, Elderly People and Picture Communication, Faculdade Motricidade Humana.

Rocha, N., Cidade, C. (1993). Description of Equipment its Operation and Facilities - Portuguese ACE, RACE 2033 TeleCommunity Deliverable.

The Active User
(Some) Advantages and Disadvantages of Educational Hypertext

MARGIT POHL, PETER PURGATHOFER
Department for Design and Technology Assessment
University of Technology Vienna, Möllwaldpl. 5, A-1040 Vienna, Austria

The active participation of readers/authors is generally supposed to be one of the major advantages of hypertext systems (e.g. Landow, 1992). This is especially important for educational systems as there is some agreement that active involvement in learning leads to better results than traditional styles of instruction (Cunningham, Duffy, & Knuth, 1993). We therefore asked the students attending lectures at our department to develop their own hypertext documents. The students had technical as well as conceptual problems. To overcome their difficulties we decided to develop our own hypertext authoring tool. Many of the students' technical problems can be solved by this system, but some of the conceptual problems still remain. We have found out that there are basically three different areas which have to be considered when creating a hypertext authoring tool for inexperienced users: structure, text, and layout.

Many students apparently found it difficult to give their documents a clear *structure*. Especially the development of meaningful non-hierarchical links seems to be a complicated task. At an early stage of our project, most of the students made no non-hierarchical links at all. After the introduction of our authoring tool, link-making became much easier and students made many more links. But these links are very often quite irrelevant for the topic discussed at the source node.

The production of *hypertext* and of traditional linear *text* are two different tasks. In an essay, an argument can evolve over several pages or even chapters. In a hypertext document, text has to be self-contained and fragmented. It is necessary to be rather short and precise, and to restrict oneself to the essential points of a topic. This can be seen as an advantage of hypertext because it makes authors think in terms of structure. On the other hand, the fragmentation in hypertext documents can also be considered a disadvantage. Users of hypertext might learn to see information only as small separate chunks of text without any relation to a larger context. Students apparently found it difficult to adapt to the fragmented and self-contained nature of hypertext.

Some of the students had problems to create an acceptable *layout* for their documents. Generally, there is too little awareness of the importance of good user interfaces. In the context of hypertext, this is all the more detrimental. A good layout can convey much of the necessary structural information to give readers a comprehensive overview about the information available. In the documents we have analysed, good structure and good layout usually go together.

The authoring tool provides means which can be used to overcome these difficulties. There is an overview editor which not only gives students feedback about the structure of their documents but enables them to edit the structure of their documents immediately. To aid students in the link creation process we use "Typed links". "Typed links" are supposed to offer students outstanding link types they never thought of before and help them avoid irrelevant links. The authoring tool provides certain standard layout features to enable students to concentrate on the text. In the last version of our system we also included an automated monitoring system which can assist students and staff of the department in the analysis of the hypertext authoring process.

References

Cunningham, D.J., Duffy, T.M, & Knuth, R.A. (1993). The Textbook of the Future. In: C. McKnight, A. Dillon, & J. Richardson: Hypertext - a psychological perspective. Chichester, West Sussex: Ellis Horwood Limited, 19-49.

Landow, G.P. (1992). Hypertext - The Convergence of Contemporary Critical Theory and Technology. Baltimore, Maryland: The John Hopkins University Press.

Acknowledgements

Our research is part of a project funded by the Austrian "Fonds zur Förderung der wissenschaftlichen Forschung", project no.: P8565-TEC. This project is directed by Prof. Peter Fleissner.

Evaluation of a HyperCard-Based Application to Enhance ESL Interactions

Merce Pujol
English Department
Hostos Community College, City University of New York
Bronx, New York, 10451, USA

Although there has been a recent explosion of Hypermedia and Multimedia related applications in the last five years, and there are a great number of articles and books on the market about the advantages and disadvantages of such devices for education (Jonassen & Mandl, 1990; Ambron & Hooper, 1988, & 1990), we have little empirical research which shows the effectiveness of such applications for educational purposes. In this study, four pairs of Hispanic intermediate ESL (English as a Second Language) learners explored a HyperCard-based software application which was developed to generate learners' oral interactions in English. Users were audio and videotaped on three different occasions while exploring the application to see to what degree they interacted with each other in English while they interacted with the software.

Description of the Software

This HyperCard software application consists of a 12-unit package where English is presented through dialogues in everyday contexts. There are four main activities in the software:

- the lesson, where users listen to and read a dialogue in context and learn the specific vocabulary of the setting:
- the vocabulary test, where users are questioned on the vocabulary learned previously;
- the writing of the dialogue that allows users to type and print their own version of a dialogue;
- the scrambled dialogue game, where users are supposed to put a scrambled dialogue into the correct order.

Findings

The four different activities proved to be very different as far as English interactions are concerned. In activity 1, the lesson, users were highly engaged in navigating the application and exploring the different options built in the software, but little oral talk occurred. When talk occurred, it was mainly procedural and in Spanish. In activity 2, the vocabulary test, users' interactions with the software were clearly merged with their negotiation of meaning with one another. Users needed to discuss the answers with each other in order to successfully interact with the software. Their interactions, however, were reduced to one or two isolated words in English. This was due to the nature of the activity, which did not require users to create more sophisticated structures in English. In activity 3, the writing of the dialogue, users mainly interacted with each other to reproduce a dialogue in English. This activity produced the most English use. Users continuously rehearsed the dialogue to be typed and read aloud what they had written in English. In activity 4, the scrambled dialogue game, users interacted with the text on the screen by arranging the boxes of text correctly on the screen. Their talk was basically procedural and in Spanish in this activity.

Conclusions

Hypermedia and multimedia computer applications can be used to create communicative activities for ESL learners. However, users should not simply navigate these programs; they also need to take an active role in the production, oral or written, of long stretches of discourse in order to carry out a specific task at the computer.

References

Ambron, S. & Hooper, K. (1988). *Interactive multimedia.* Redmond, WA: Microsoft Press.
Ambron, S. & Hooper, K. (1990). *Learning with interactive multimedia.* Redmond, WA: Microsoft Press.
Jonassen, D. H. & Mandl, H. (Eds). (1990). *Designing hypermedia for learning.* Berlin: Springer-Verlag.

Software Tutorials vs. Instructional Video Sessions for Introducing Educational Technology to Teachers in a Chilean Elementary School Network

LUCIO REHBEIN, ARNOLDO VIDAL, MIGUEL RIPOLL, PEDRO BARRIENTOS and PEDRO HEPP
Projecto Enlaces
Universidad de La Frontera, Casilla 380, Temuco, Chile

The school network which is presently in growth, began with 6 school-nodes in Santiago during 1992 (see Hepp, et al., 1993), and it was then given full support and a decentralized implementation in Araucanía, one of the poorest and educationally worse-off regions of the country. The project, best known as "Proyecto Enlaces" represents the first official attempt ever done by the State of Chile, through its Ministry of Education, to generate new and pertinent answers to questions pertaining the role of technology, particularly that of computers and communications, in improving the quality and equity of education in the Chilean elementary school system. Until now, once admitted in the network, teachers have received on-site face-to-face training on each software application put at their disposal.

Our cumulated data thus far indicates that, before training began, 25% of the teachers thought they were incapable of ever learning to use a computer; 23.3% believed that, given their cost, computers would never become part of their schools; 18.7% believed computers were meant for any other use but education; a similar proportion of teachers (18.5%) feared that if he/she made a mistake, the machine would brake apart or explode; and, a 12% of them feared computers could take over the role of the teacher. Only 0.2% had prior knowledge and experience with computers.

Two months after training, 97% of participating teachers declared having developed a significantly more positive attitude towards computers than the one they had at the beginning, and 70% of them felt that, even though they needed considerable more practice, they could successfully use the computer in support of their regular teaching requirements.

In quest for a new approach to teacher training

The training scheme which has just been discussed proved to be beneficial while there was just a dozen schools in the network, and all of them located within a reasonable geographical distance. The planned expansion of the network for 1994 will mean at least a fourfold increase in the number of schools, and quite probably a similar increase in the number of teachers in need training. Furthermore, new schools will be distributed an area of approximately 31,850 Km^2. Any attempt to give live, on-site, teacher training in this new scenario, would be completely impossible.

From these new conditions, there derives an extreme need for implementing a new teachers training program: One that would not require the presence of specialized professionals for it to be successful and, at the same time, one that would retain the quality and avoid the errors of the previously implemented program. Two proposals for such a program are briefly sketched along the following paragraphs. Both programs will be put into test during the next expansion of the network in march, and their effectiveness assessed and compared, by may 1994.

Briefly, this proposal implies to compare, under as similar conditions as possible, the effectiveness of two self-administered instructional packages: One, consisting in an interactive software developed as a Macromedia Director application. The other, consisting in an instructional video containing exactly the same information contained by the software, but with emphasis on referencing actual recorded information from previous training programs.

References

Hepp, P., Alvarez, M.L., Hinostroza, E. and Laval, E. (1993) "La Plaza": A Software Design for an Educational Network. In H. Maurer (ed.) *Educational Multimedia and Hypermedia*. Charlottesville VA:: AACE

Research on conventional dictionary usage as insight for the design of online dictionaries in scholarly workstations

Warren B. Roby
Foreign Languages and Literatures
Washington State University, Pullman, WA 99164-2610 USA

Dictionaries are integral to hypermedia workstations (Weissman, 1988). These online resources would seem to be especially welcome in areas such as foreign language study where dictionary use is heavy. However, one cannot automatically assume that more resources guarantee more learning. This paper summarizes studies of the use of both conventional and computer dictionaries and glosses in foreign language learning. It is argued that the conclusions drawn from them may inform the design of online dictionaries on academic workstations.

Benoussan, Sim, & Weiss (1984) concluded that "less proficient students lack the language skills to benefit from the dictionary, whereas more proficient students know enough to do without it." (p. 271). Lantolf, Labarca, and den Tuinder (1985) identified three three usage strategies: an avoidance strategy, a semantic field strategy, and a lexical formalism strategy. Aust, Kelley, and Roby (1993) found that subjects using computerized versions of dictionaries looked up significantly more words ($F= 26.96, p<.000$) than those using the paper versions of the same dictionaries. In a followup study, Roby and Aust (1994) replicated this finding and also found that subjects who had text-specific glosses in addition to the dictionary read significantly faster ($F=4.62$; $p=.034$) than subjects who only had access to a dictionary (be it paper or computerized). No comprehension differences between the various kinds and formats of dictionaries were discovered in the Aust and Roby studies, but subjects preferred bilingual versions to monolingual ones. Davis (1989) found that subjects who received glosses understood significantly more of a passage than those who had no reading support.

Book dictionaries are said to be "arduous" to use, whereas on-line dictionaries are supposedly fast and easy (Keller, 1987). If such is the case (evidence to that effect has been cited above), it may be that users employ different strategies in their interactions with these aids depending on the media they are in (Daniel & Reinking, 1987). There is almost certainly a novelty effect for many users of computerized aids; this suggests that advisement needs to be built into online dictionaries to encourage their proper usage. Learners need to be counseled by teachers and reminded by online systems that a lexical "quick fix" is no substitute for careful reading and inferring from context. Although both bilingual and monolingual (e.g. Spanish-to-Spanish) dictionaries and glosses can exist online, language learners need to be nudged away from overdependence on the former.

References

Aust, R., Kelley, M. J., & Roby, W. B. (1993). The use of hyperreference and conventional dictionaries. *Educational Technology Research and Development Journal*, 41, 63-73.

Benoussan, M., Sim, D. & Weiss, R. (1984). The effect of dictionary usage on EFL test performance compared with student and teacher attitudes and expectations. *Reading in a foreign language, 2*, 262-276.

Daniel, D. B. & Reinking, D. (1987). The construct of legibility in electronic reading environments. In D. Reinking (Ed.) *Reading and Computers: Issues for theory and practice* (pp. 24-39). New York: Teachers College Press.

Davis, J. N. (1989). Facilitating effects of marginal glosses on foreign language reading. *Modern Language Journal, 73*, 41-47.

Keller, H. H. (1987). Pedagogical wishes for a machine dictionary: An example from Russian. *Modern Language Journal, 71*, 12-17.

Lantolf, J. P., Labarca, A., & den Tuinder, J. (1985). Strategies for accessing bilingual dictionaries: A question of regulation. *Hispania, 68*, 858-864.

Roby, W. B. & Aust, R. (1994). Glosses and dictionaries in paper and computer formats as adjunct aids to the reading of Spanish texts by university students. Unpublished Manuscript.

Weissman, R. F. E. (1988). From the personal computer to the scholar's workstation. *Academic Computing, 3*, 10-14; 30-41.

Hypermedia project: Audio, video, and HyperCard combined to improve student strategies for foreign language comprehension

K. SCHAIRER
Department of Modern Languages
Northern Arizona University, Flagstaff, Arizona, 86011, United States

The Input Hypothesis, an integral component of Krashen's Natural Approach to language acquisition, contends that conversational fluency is acquired when learners are exposed to and understand verbal messages that are a little beyond their current ability. Language is "picked up" from target language input which is comprehensible enough for the listener to connect meaning to new linguistic forms through context. For second language acquisition to take place, both texts and tasks must be matched to the listeners' abilities.

A 1984 Texas study finds that 74.8% of college seniors with second language degrees achieve an Intermediate High rating on the ACTFL scale (Hiple & Manley,1987). This means they leave the learning environment dependent on simplified speech, without sufficient receptive skills to interact comfortably with native speakers in a natural environment or to benefit from authentic language input provided by media. If graduates are to avoid stagnation of their communicative skills upon leaving the classroom, they must develop comprehension strategies which allow them to take advantage of authentic language in the real world.

ACCESS, an interactive audio program designed to help second language students develop comprehension strategies, uses unsubtitled films as the source of input. It interrelates key sequences recreated in audio and text form, vocabulary helps, cultural explanations, and comprehension questions with feedback. The aural segment, temporarily accessed through tape players, will be script-linked upon resolution of program/hardware incompatibilities. The program will serve as a template for use with other foreign language films whose content is beyond student skills. The pilot project is based on a film version of the Argentine epic poem Martín Fierro which evolves around a single character, maintains a limited point of view, and focuses on a single sequence of events narrated in chronological order. ACCESS is tailored to deal with the specific regionalisms and unfamiliar cultural material to be expected in unsubtitled foreign films.

In the pilot study, Martín Fierro was viewed by thirty second-year and twenty fourth-year Spanish students who wrote questions after the first viewing which they answered after the second. Between viewings, half of each group explored the computer program and half participated in class discussion.

Initial results show surprisingly little difference in expressed frustration and incomprehension among second and fourth-year students after their first exposure to Martín Fierro, though questions written by the latter indicate much greater actual comprehension. Plot summaries written by students who spent time on the computer are less accurate but more individual than essays written by students who participated in discussion groups. Self-evaluation of comprehension based on a 1-5 scale indicates that second-year students felt they improved by an average of 1.4 points, and fourth-year students rated their average improvement at .9.

Text reduction through cloze activities and the addition of audio input, even under less than optimum circumstances, significantly improves student evaluations. Text alone rates 3.6 whereas the combination of text and audio rates 4.4. Cloze exercises rate 4.6. Self-evaluation of comprehension does not increase significantly with access to audio, but whereas only 20% of general comments are positive from the groups accessing computer text only, 80% are positive when students can see the text and control the audio.

Students tend to consider unsatisfactory anything less than complete comprehension. Preparation for real-world comprehension tasks must include increasing student tolerance for the unknown and decreasing frustration thresholds. Student satisfaction and the reduction of frustration levels are more significant than measures of comprehension when the goal is an increase in voluntary exposure to authentic language input.

References

Hiple, D.V. & Manley, J.H. (1987). Testing how well foreign language teachers speak: A state mandate. *Foreign Language Annals, 20*, 147-153.

Krashen, S.D.& Terrell, T.D. *The Natural Approach*. Oxford: Pergamon.

Acknowledgements: The ACCESS project was made possible through an Instructional Development Grant from the Office of Professional Development, Northern Arizona University, Flagstaff, Arizona.

The Design of an Effective Software Interface for Interactive Video

PENELOPE SEMRAU, PH.D. AND BARBARA A. BOYER, PH.D.
California State University, 5151 State University Drive, Los Angeles, California, USA 90032-8143

In this paper, guidelines for designing an effective software interface for interactive video are examined. At a basic level, *software interface* is defined as the communication system between the computer and the user. At a higher level, interface is the process whereby the learner interacts, participates in, and contributes to the learning event (Rieber, 1994, 211). The design of the interface can significantly effect the quality of interaction that the learner has with the learning environment as well as the construction of one's own knowledge base. From a constructivist view, one's knowledge is constructed while interacting with the learning environment. Thus, the design of the interface with which the learner interacts impacts the kind and quality of knowledge and learning experiences derived from this interaction.

Interactivity, cueing strategies, global control, local control, learner control and pacing, use of symbols for navigating, and the use of consistent applications are interface design factors that are examined for their relationship to interactive video.

Interactive video should allow students to select from various options and to branch, as well as to create their own material--demonstrating imagination, critical thinking, and problem-solving skills. If the interactive video does not attempt to provide directions for students, be concerned with levels of student achievement, or account for different individual learning styles, then whatever interactivity exists is of limited worth in an educational setting (Semrau, P. & Boyer, B. A., 1994). Passive video viewing, lecture formats, the use of drill and practice, and student-chosen paths without direction or feedback tend to be the poorest use of interactive video (Yoshii, Milne, & Bork, 1991).

The issue of how to relate abstract, formal knowledge to particular real world situations can be addressed by the interface design. Students have difficulty linking their abstract understanding to actual applications. Being able to alter and manipulate microworlds in interactive video programs allows students to use both formal cognitive information to applied knowledge in the sciences, the arts and humanities.

References

Rieber, L. P. (1994). *Computers, graphics, & learning.* Dubuque, IA: Wm. C. Brown Communications, Inc.

Semrau, P. & Boyer, B. A. (1994). *Using interactive video in education.* Boston, MA: Allyn & Bacon.

Yoshii, R., Milne, R. & Bork, A. (November, 1991). Highly interactive programs with video for learning languages. In *Proceedings of the 33rd Annual International Conference of the Association for the Development of Computer-Based Instructional Systems*, St. Louis, MO.

Authoring Considerations in the Imaginary Museum Project: The Sistine Chapel A Case Study

K. SEQUERRA, A.R. ROCHA, N. SANTOS
Universidade Federal do Rio de Janeiro / COPPE
P.O. Box 68511 E-Mail: Sequerra@COS.UFRJ.BR

S. OLIVEIRA
Pontifícia Universidade Católica do Rio de Janeiro
Marques de São Vicente 255 - Rio de Janeiro - RJ - Brazil

The Sistine Chapel Project has been created based on Andre Malraux's (Malraux, 1947) conception of a virtual, or imaginary museum. Its basic idea is that given the invention of photography and mass production of picture books, it is no longer necessary to travel to see a museum. Each book containing a reproduction of a work of art is itself a virtual museum. The project extended this conception from books to multimedia. Among the many educational goals envisioned by it, the most important ones concern the integration of cultural and artistic aspects into a regular curriculum scenario. The inherent complexities involved demanded the use of two authoring methods. Both authoring-in-the-large and authoring-in-the-small approaches had to be used to model the application. The methods chosen were HDM (Garzotto, 1990) and Hiper Autor (Breitman, 1993) respectively. The authoring-in-the-large approach allowed us to create a framework where other applications can be developed, e.g. The Brazilian Museum of Folklore, while the authoring-in-the-small approach helped us in modeling the Sistine Chapel application in particular.

The conjunction of both methods proved to be a very useful strategy. The combination of the advantages offered by both approaches have resulted in the production of a high quality and reliable application that is being used now by students at schools. Among the benefits are:

completeness, in the sense that it is possible to model the application in two levels of abstraction, given by authoring-in-the-large and authoring-in-the-small, providing total understanding of the application's objectives while avoiding inconsistencies and mistakes,

expression, allowing the description of concepts and structures at the appropriate level,

simplicity, so that the methods can be used by any authoring team, independent of their previous experience in developing hypermedia applications (Garzotto, 1991; Breitman, 1993, Alty, 1993).

We believe that in the near future we will continue to use the strategy derived from the use of both methods to underline the construction of new applications in the imaginary museum context. Among the possibilities are the production of an application concerning the works of art in the Brazilian Museum of Modern Art.

References

Alty, James (1993) - We have the technology but do we have a methodology? - *Proceedings of ED MEDIA 1993* - 1993

Breitman, Karin K. (1993) - *Hiper Autor: a method for specification and project of hypermedia systems* - Thesis - COPPE/ Federal University of Rio de Janeiro - September 1993.

Garzotto, Franca; Paolini, Paolo; Schwabe, Daniel (1990) - *HDM: a model based approach to hypertext application design* - Technical Report -Politecnico de Milano - 1990.

Garzotto, Franca; Paolini, Paolo; Schwabe, Daniel ; Bernstein, Mark (1991) - Tools for designing hyperdocuments - *Hypertext Hypermedia Handbook* - McGraw Hill - 1991

Malraux, André (1947) - *Le musée imaginaire* - Gallimard - 1947.

Acknowledgment: The authors would like to thank professor Daniel Berry for his contributions to this article.

Interactive Video in Preservice Physical Education Instruction: Comparing Two Interaction Approaches

Eric E. Smith
Division of Research Evaluation and Development
University of Northern Colorado, Greeley CO 80639

Del Engstrom
Ursinus College, Department of Exercise and Sports Science
P.O Box 1000, Collegeville, PA 19460-1000

The purposes of this study were twofold. The first purpose was to develop an interactive video system designed to introduce preservice teachers to potential discipline issues experienced in physical education. The second purpose of the study was to determine the impact of an interactive video system and group decision support systems technology on the acquisition of discipline content knowledge. Specifically, the study examined: (a) the effectiveness of an interactive video system in acquiring discipline content knowledge; (b) the effectiveness of group decision support systems technology in acquiring discipline content knowledge; and (c) the effectiveness of interactive video in combination with group decision support systems technology in acquiring discipline content knowledge

Subjects were 55 college students enrolled in two Analysis and Movements of Volleyball courses during the 1993 spring semester at the University of Northern Colorado. The subjects were assigned to one of three treatment groups based upon two characteristics--gender and pretest scores on a Discipline Content Knowledge Test

The interactive video (IV) group consisted of six females and twelve males received instruction in discipline content knowledge using interactive video. The decision support (GDSS) group viewed the discipline scenarios using the group decision support laboratory. Six females and twelve males comprised the second group The combined (COMB) group consisted of seven females and twelve males who received instruction using interactive video and the group decision support laboratory.

Results

A one-way analysis of covariance (pretest for content knowledge was the covariate) indicated no significant differences among the three treatments groups on posttest scores, $F(2,51) = 0.33$, $p = 0.717$. However, all three groups did show improvement from pretest to posttest with only the IV group showing a significant increase in discipline content knowledge as shown by t-test, $t=-2.81$, $p=0.006$. Both the GDSS and COMB groups required 135 minutes to complete all six scenarios, while the IV group averaged 45 minutes with range from 32.6 to 65.6 minutes.

From these results, it is evident that the IV treatment is at least as effective as the other two treatments In addition, the IV group completed the material in one third of the time and without instructor intervention This time savings and the fact that the material was learned without the instructor suggests that implementation of this type of instruction will reduce training time and provide opportunities for instructors to either cover more material or cover material to a greater depth, concentrating on problem solving or other "high level" skills that benefit from instructor intervention

Designing Effective Multimedia: Guidelines for the Interface

Eric E. Smith
Gayle M. Munson
Thomas Lightner
Joseph Gregg
Division of Research, Evaluation, & Development
University of Northern Colorado, Greeley CO 80639

Multimedia is fast becoming THE buzzword of the interactive training world leaving both consumers and developers of interactive training confused by the vast array of widely differing products and options with little guidance for selection or implementation. Practitioners involved in developing multimedia computer-based interactive training systems are currently producing their systems in a standardless world. The lack of standards creates more uncertainty for students, end users, and novice developers than is desirable. For example, students may be unfamiliar with icons that developers take for granted. The "return" icon may be confused with the "previous page" icon. In addition to the possible unfamiliarity with the interface conventions used, the students also bring their own preconceived notions of how a system should work. When the interface and the student's mental model match, the student may find the system easy to use and effective. When there is no match, the student may find a system difficult and possibly frustrating to use. In general, the goals of interface design for the instructional/educational use of multimedia are: an environment that supports learning; a system that does not frustrate the user; and a system that allows the user to accomplish learner goals. A set of guidelines would be helpful in meeting these goals. However, Shneiderman (1992) cautions that "A clever design for one community of users may be inappropriate for another community An efficient design for one class of tasks may be inefficient for another class" (p. 12). Our search for guidelines examined information structures, navigation structures, and icon use.

There are two general types of information structures for multimedia design the physical structure for the development environment and the cognitive information structure (concept map) While a number of different ways to analyze and design these structures have been identified, there is little empirical guidance for the effective use of the structures. Several studies have investigated navigation structures but, the results appear to be user purpose and context dependent. Little is known and even less has been investigated concerning icons used for instructional/educational multimedia Icons should be intuitive representations of the object or action for which they stand Where confusion may arise, the icon should be replaced by either a different icon or unambiguous words

In searching for a "better way" to develop multimedia documents for learning/instruction, many offers of guidance have been found, but few with any empirical basis. For the most part, guidelines derive from two sources, theory (whether untested or based on what works for individual media) and experience While experience can often lead in the right direction, it can just as often lead in the wrong direction Those studies that are reported suggest that many of the techniques tested made little or no difference in learning performance However, the short duration of the treatments in many studies may provide misleading results Only with long duration treatments can the actual impact of multimedia designs on learning and attitudes be determined Until validated guidelines for the design of hypermedia and multimedia documents are developed, the facilitation of the learning process will be hit or miss at best While the available, largely untested, guidelines are useful, they should be used with a "grain of salt" To be effective, any interface must be thoroughly tested Where the guidelines are accurate, follow them Where they are not, experiment and test using your audience until an adequate solution is found

References

Shneiderman, B (1992) Designing the user interface Strategies for effective human-computer interaction Reading, Massachusetts Addison-Wesley Publishing Company

So What If It's In Color and Moves? A Critique of Multimedia

MICHAEL SZABO Ph.D.
Department of Adult, Career and Technology
University of Alberta, Edmonton, Alberta, T5G 2H3, Canada

BRENT POOHKAY, 2nd author
Faculty of Law
University of Alberta, Edmonton, Alberta, T5G 2H3, Canada

Educators have used the term multimedia in a global sense for years: outside the classroom the term was rarely heard. Research in multimedia was spurred after the second World War by the military interested in increasing training effectiveness and conducted by new research graduates eager to carry out the studies.

In recent years, the prospects of huge businesses to be spun out of the multimedia and information superhighway efforts has brought the term multimedia out of the closet and into the public view. For example, Naisbitt (1994) has written that multimedia will become a dominant force in the next decade and the two most significant applications will consist of entertainment and education.

Multimedia may be defined as the components of conventional media (computer, video, graphics, animation, audio, color) along with their path to complete integration, which is now just in its infancy.

This paper examines research and guidelines for use for the multimedia components of color and animation. Color was chosen because there is so much inconect information in common folklore about the alleged effects of color on achievement. Animation is examined as it is the new kid on the block which will likely grow quickly as the hardware and software systems bring us desk top animation. The paper will conclude with a summary of a research study which demonstrates positive learning effects of animation on learning and attitude.

Most instructors would agree that color used in instruction (1) increases learning, (2) makes the instructional environment more appealing and therefore increases attention and motivation to learn, and (3) does not distract from the learning tasks at hand Unfortunately, they are not completely correct. Extensive research on using color to increase the level of reality in an instructional situation shows that learning is not generally enhanced by color (Wise, 1982; Dwyer, 1967, 1968, 1970), with certain exceptions. Several recommendations for the use of color will be provided.

Animation refers to the use of a series of graphics which change over time or space. There is a significant body of research on graphics in instruction in training with considerable evidence as to their effectiveness. Although animation does not have the rich research history that is associated with color and graphics, the results are encouraging. For example, Baek and Layne (1988) compared learning conditions of text only, text plus graphics, and text plus animation. The adults in the study scored higher in the animation condition than either text or graphics. The animation condition also resulted in less study time, suggesting that animation results in more efficient learning. In another study with adult learners, Mayton (1991) found increased scores in the animation condition immediately after study persisted and were measurable one week later. The authors recently found that animation results in increased achievement of the learning task and improved attitude.

A major recommendation presented in this paper is to analyze the relevance of graphics/animation and color cues to the learning outcome and use those cues appropriately in the instructional, practice, and testing situations relative to the particular learning objectives.

Knowledge Based Tutoring System for Learning Lithuanian Language

K.BANIULIS, B.TAMULYNAS
Department of Computer Software
Kaunas University of Technology, Studentu 50-402, 3028 Kaunas, Lithuania

D.DMUCHOVSKA, J.NORMANTIENE
Department of Lithuanian Language
Kaunas University of Technology, Gedimino 43-204, 3000 Kaunas, Lithuania

The purpose of this work is to build up Computer Based Lithuanian Language Learning System which will satisfy needs of various users (school-children, students, etc.). The Computer Based Lithuanian Language Learning System is considered as a part of the general Intelligent Tutoring System (ITS), which include the following modules: subject oriented tutoring modules, subject oriented data base modules (vocabularies, tables, specific subject information), tasks and lessons making data base modules, and knowledge based students modules. The project of ITS for learning Lithuanian language includes such steps: 1) building up the vocabulary; 2) filling in the vocabulary; 3) making lessons; 4) learning process control.

The main goal of the computerized vocabulary is to accumulate and supply systematized and classified information about words and their constituent parts. Commonly, a word can have more than one constituent part of the same type. On the other hand, the same constituent part can be a member of a great variety of words. Therefore, to avoid redundancy we have used a relational form of the vocabulary. Within the limits of the School Orthographic Dictionary of Lithuanian (about 16000 words), classification, structurization and characterization of information presented in the dictionary was made. Classification means assigning each word to one of the parts of the language (e.g. Noun, Verb, etc.). Structurization means singling out syllables and parts of a word and indication of stress. Characterization is connected with indication of certain attributes and additional information, and is performed for each of the parts of the language according to a separate pattern. The Lithuanian language model implemented in the vocabulary base allows to perform syntactical and morphological analysis of student answer (Baniulis, Dmuchovska, & Tamulynas, 1993).

The tutor designs lessons using vocabulary and task bases, forms his own teaching strategies preparing compound lessons. The teacher can use an ability to add new subject knowledge into the data bases if necessary. Traditional exercices, game situations and dynamic models are being employed for the lessons modules, using various techniques: to answer, fill in the gaps, compose the answer from separate elements. In the game-play approach traditional tasks are supplemented with visual illiustrations (a train, stairs, a balloon etc.). Dynamic models make the original graphic illustration of the verbal text. The special authorized editorial programs and modules are used to make easy the design process of all types of these lessons.

The ideal behaviour of an ITS system for knowledge based learning process control is required to be similar with that of human tutor. This purpose is achieved by making references to a human tutor's desired characteristcs and simulating natural learning process. ITS consists of the four types of knowledge based modules. They are the problem-solving expertise, the knowledge based student model, the tutoring strategies and the model of natural interaction process student-computer.

All these ITS's modules as well as another constituent parts and data bases are implemented in C++ object-oriented manner. So, they are closely interconnected with each another and have posibility for corrections and additions. Currently the ITS for learning Lithuanian language is on going research under testing and verification presented modules and data bases. We hope the first version of this system soon will be used in Lithuanian schools and universities for training school-children or teaching foreign students.

References

Baniulis, K., Dmuchovska, D, & Tamulynas, B. (1993). Computer-based tutoring system for learning Lithuanian language. In *Proceedings of the International Conference on Computer Technologies in Education,* Kiev.

Changing the Way Electromagentics is Taught: Precision Teaching + Interactive Simulations

C. THOMPSON, L. HODGES, J. MARR, N. WALKER, E. THOMAS, A. CABRERA, A.THOMAS, J. WOOD
Departments of Physics, Psychology, and GTRI
Georgia Institute of Technology, Atlanta, GA, 30332

An introductory physics course is taken by approximately 100,000 pre-engineering students in the United States each year. More than 30% of these students fail to graduate as engineers. Most of that attrition occurs during the first two years, in part because of struggles with basic mathematics and science courses. The present effort focuses on developing and assessing computer-based instructional systems for enhancing student performance in the second-quarter physics course in electricity and magnetism. This course is perceived as the most difficult of a three-quarter sequence, with some 30% of students making 'D', 'F', or withdrawing.

Part of the difficulty of the course resides in erroneous notions about concepts such as voltage, current, and electrical energy brought to the course by students [3]. While a basic course in mechanics deals largely with forces and moving objects accessible to everybody in everyday life, electricity and magnetism are phenomena generally accessible only through complex measuring devices. Thus, establishing intuitions about such concepts is a hard task when only traditional materials are available. Another problem identified by veteran instructors and a diagnostic test is a less than automatic application of basic mathematics. Students spend a great deal of cognitive energy executing routine calculus procedures.

Two very different instructional strategies are currently being applied to these problems, precision teaching of basic skills and interactive simulation. Precision teaching, a concept pioneered by Lindsley [4] is a technique for the enhancement of basic skills and achieving fluency. Students undertake a large group of simple problems and their success is measured by the number of correct responses in a fixed time interval. Fluency of correct responding, or the rate of correct responses, is a critical feature of the evaluation [1]. With repeated exposure to different problem sets, rate of correct responses typically increases. The rate is actively recorded by the student. It has been demonstrated that using this approach basic skills are enhanced, performance in standard courses is improved and the improvement is retained for extended periods [2]. A prototype exploratory simulation microworld, Electroworld, is currently under development as well. This environment allows students to interact with forces, fields, and charges in a concrete visual manner. As a result of engaging this simulation environment students are learning to visualize many abstract concepts of electromagnetics for the first time. Preliminary analysis of a group of students using the precision teaching material as compared to a more traditional approach of assigning additional homework problems has shown a significant shift in final grade distributions from D and F to A and B.

References

1. Binder, C. V. (1988). "Precision Teaching: Measuring and attaining academic excellence."*Youth Policy*, 10, pp. 12-15.
2. Binder, C. & Watkins, C. L. (1990). "Direct instruction: Measurably superior instructional technology in schools. *Performance Improvement Quarterly*, 3, pp. 74-96.
3. Carlsen, D. C. & Andre, T. (1992). "Use of a microcomputer simulation and conceptual change text to overcome student preconceptions about electric circuits." *Journal of Computer-Based Instruction*, 19, pp. 105-109.
4. Lindsley, O. R. (1972). "From Skinner to precision teaching." In J. B. Jordan & L. S. Robbins (Eds.), *Let's try doing some other kind of thing*, (pp 1-12). Arlington, VA: Council on Exceptional Children.

Computer Based Instruction in Traffic Theory

KAREL VAN DEN BOSCH
TNO-Human Factors Research
PO Box 23, 3769 ZG Soesterberg, the Netherlands

Traffic legislation is the primary source for deriving education in traffic theory in the Netherlands. Consequently, programs are formal and abstract, emphasizing definitions, the meaning of signs, and right-of-way rules. For example, students learn to identify the sign indicating a motor-way, learn that a car belongs to the category of motor-vehicles, and that the speed-limit for motor-vehicles on motor-ways is 55 miles/hour. It is up to the student to acknowledge the implications of the newly acquired knowledge for driving behavior in actual task situations. Thus, the relation between theory and practice is very indirect. This approach appears to fail at teaching students the necessary knowledge and skills for driving a car, as indicated by a non-existent or low contribution to driving performance (Brown, Groeger, & Biehl, 1987). It is our belief that if instructional material is selected on its relevance for every day traffic behavior, education in traffic theory càn contribute much more to the development of proficient driving.

This paper reports the development of a program for teaching traffic theory, based upon principles of learning and instruction. Recent technology offers ample opportunity for effective application of these principles. The program is therefore designed for use in a multi-media environment. The project is carried out under contract to, and in collaboration with, the driving schools of the Royal Netherlands Army.

Cognitive psychology has shown that the prerequisite knowledge and skills to perform a certain task are best attained if the material to be learned is selected according to its relevance for actual task behavior, and presented in a representative context ('situated action', e.g. Norman, 1993). The present program meets these principles by giving instruction on how to recognize traffic situations, how to identify potential problems and conflicts, and how to act appropriately when in such situations. To illustrate for the example presented above: students are instructed how to recognize motorways (on formal, but also on informal characteristics, like dual carriageway) and how actual task behavior is affected by various factors (such as speed limit, road- and weather conditions, traffic intensity, etc.).

The situated action framework indicates that students can make the link between theoretical knowledge and practice more readily if the information is presented in real-life like contexts. The efficacy of instruction can be further increased by comprehensive student activity. Recent technology is utilized to accomplish these objectives. For example; digitized photo's offer new possibilities for instruction in road classification; features of a road category can be added or deleted at will, thus allowing for effective highlighting of essential characteristics. Defining sensitive area's in the picture permits program-student interaction (e.g. by mouse-clicking). Until recently, dynamic aspects of driving could not easily be implemented in interactive learning situations. The present program utilizes digitized video to show (potential) problems and conflicts in traffic situations, to show the outcomes of different reactions to a certain problematic situation, and to show the antecedents that produced the problematic situation in the first place.

The development of the program is currently underway along the lines presented above. Two modules (of 12 in total) have been developed in full in a prototype. An evaluation that was subsequently carried out produced very promising results. Especially the video fragments turned out to be very instructive and motivating. Instructional materials for the other modules will be implemented in an inclusive program of which we hope to show that it optimizes actual driving performance by providing students with comprehensive and relevant knowledge and skills.

References

Brown, I.D., Groeger, J.A., & Biehl, B. (1987). Is Driver Training Contributing enough towards Road Safety? in: J.A. Rothengatter, & R.A. de Bruin (Eds.), *Road Users and Traffic Safety*, (pp. 135-156). Assen: Van Gorcum.

Norman, D.A. (1993). Introduction to the Special Issue on Situated Action. *Cognitive Science*, *17*(1), 1-7.

MNP: A Multimedia-Based Project to Teach an Introductory Nutrition Course

JOAQUIN A. VILA* and CONSTANCE G. MUELLER**
*Applied Computer Science and **Home Economics
Illinois State University, Normal, IL 61790, U.S.A.

The Multimedia Nutrition Project (MNP) was designed and developed to improve the delivery of nutrition instruction by means of multimedia technology. MNP consists of a series of modules which are to be used in both lecture presentations and student labs. Some of the specific topics include dietary behavior modifications such as low fat food choices and low fat cooking methods, dietary guidelines, food guide pyramid, dietary fat and heart disease, and composition of foods. Each module had two levels of interactivity: 1) multimedia classroom presentation (MCP) used by faculty members to deliver instruction of a selected topic and 2) multimedia labs (ML) used by students for drilling and testing of concepts introduced in lectures.

MNP Modules

The following are some of the module highlights of MNP.

1. The MCPs were developed in Freelance Graphics 2.0 for Windows. Many link other customized applications created in Toolbook, Excel, 3D Studio Animations, and Digital Video Interactive (DVI). For example, one MCP embedded a graph produced in Excel depicting the fatty acid composition of oils in foods. As new formulations of products occur, the graph can be easily updated. Another interesting aspect of MCPs is the random student picker icon developed in Toolbook. When clicked, it displays name and image of a randomly selected student in the class. Another MCP focuses upon processing of fats. Hydrogenation becomes more than a word when the students view a three-dimensional animated model (developed in 3D Studio) of the fatty acid structure. The rotating carbon molecules demonstrate the difference between the cis and trans forms of the molecule. MCPs provide faculty members with the ability to incorporate relevant video material into a digital video library using DVI technology. At the touch of a button, such video clips can be retrieved and played when appropriate.

2. MLs were developed using Multimedia Toolbook, 3D-Studio, and DVI. One ML contains a variety of activities to reinforce the concept of lowering dietary fat. The activities include browsing through audio-visual material depicting appropriate meat serving sizes, low fat cuts and grades of meat as well as low fat cooking methods. Video clips pertaining to heart disease and artificial fats can be accessed to relate current research. In another ML students interactively review the U.S. Dietary Guidelines. Each of the 7 principles is presented using animation and hypertext capabilities. The student also can move through the Food Pyramid interacting with each of the food groups. Each group describes the appropriate serving size and number of servings. Voice over is used to present additional instructor comments. The student can also navigate further through the pyramid to learn about sample menus which illustrate the amounts of actual food recommended.

The purpose of this project was to design and develop an interactive multimedia system to aid instructors in teaching and their students in learning basic nutrition concepts. A pilot test involving twenty two students in an introductory nutrition course was carried out. The results from the post-evaluation were very promising. However, more research is needed to assess the impact of this method of instruction. The next step is to further explore the usefulness and acceptance of the system.

Acknowledgment
This project was funded by a grant of the College of Applied Science and Technology (CAST) at Illinois State University to promote the use of multimedia technology in the classroom.

Cost-Effectiveness in Decision Making about Media in Education and Training: An (Inter)active Decision Support Approach

J.M.WETTERLING
Faculty of Educational Science and Technology
University of Twente, P.O.Box 217, 7500 AE Enschede, The Netherlands

The main reasons that cost-effectiveness of media in education is so difficult to analyse are the different definitions of the term cost-effectiveness, measurement problems and interpretation problems. The first problem automatically causes the second which is additionally improved by differing measurement methods, criteria values and infrastructures within analysis takes place. Differences between people, cultures, systems improve the problem of using a consistent frame of reference (Kozma, 1991 and Clark, 1990). Because of these differences, assisting decision makers in the use of general frameworks should have priority then over the construction of more strictly defined methodologies that constrain the individual flexibility that is needed in personal and organisational settings which have the majority in decision making settings.

Cost analysis of educational media projects has been topic of research for a long time and there are some useful frameworks developed in the past (Levin, 1983). The effectiveness of educational media is a greater problem. Technical problems with measurement instruments, comparability and validity, and media which are often used as a part of a curriculum, make results, if available, rather questionable to draw reliable conclusions from (Strittmatter, 1992). The use of indicators of effectiveness like test score increase, change in attitudes or performance improvement and the expectation of decision makers about the potential of the media in reaching criteria values on these indicators can be a solution to reliability and interpretation problems in this area (Moonen, 1990).

To assist decision-makers in cost-effectiveness analysis an instrument has been developed. This instrument (called ESTIMA) is a computer program consisting of nine units. Three units deal with the selection and evaluation of media. Three units deal with the product level when a media selection has already been made. One unit is on resource allocation where decision makers can allocate their resources to a large set of cost items (people, equipment, etc.) and analyse also the sensitivity of certain allocations on other items. One unit deals with effectiveness estimation where expectations, empirical data and experience should be combined to come to a reliable estimation of the effectiveness of the use of a media application. The final unit deals with return on investment and here the decision maker can analyse the costs and returns of the use of modern interactive media compared to the use of traditional classroom teaching strategies. The program had to be flexible with regard to the users background, easy to learn, user friendly, well documented and not too much time consuming. The program has been developed with the Visual Basic programming environment under Windows. An evaluation of ESTIMA is planned where the value of the program will be evaluated and also the general usability of decision support systems in cost-effectiveness analysis and differences in the use of these systems between decision makers in commercial and in non-commercial organisations, between decision makers in large and in small organisations and between decision makers in different academic levels.

Clark, R.E., & Sugrue, B.M. (1990). North American disputes about research on learning from media. *International Journal of Educational Research*, 14 (6), 507-519.

Kozma, R.B. (1991). Learning with media. *Review of Educational Research*, 61 (2), 179-211.

Levin, H.M. (1983). *Cost-effectiveness · A primer.* Beverly Hills: Sage.

Moonen, J.C.M.M. (1990). *Costs, effectiveness and expectations: Concepts and issues with respect to educational technology*. Report submitted to the Educational Technology Center of British Columbia.

Strittmatter, P. (1992). European Research on Media and Technology in Education: Current Status and Future Directions. *Instructional Media and Technology Research*, 489-505.

A Decision-based Hyper-multi-media Case Environment for the Computer Productivity Initiative, a Large Undergraduate Project

Chris Wild[1], Kurt J. Maly[1], Chenglin Zhang[1], Dennis E. Ray[1], Irwin B. Levinstein[1], Stephan Olariu[1], C. Michael Overstreet[1], Nageswara S. V. Rao[2], Daniela Rosca[1], Tammy Taylor[1], Tijen Ireland[1] and Deane Sibol[1]

[1] Department of Computer Science, Old Dominion University
Norfolk, VA 23529-0162, email: wild@cs.odu.edu
[2] Intelligent Systems Section, Center for Engineering Systems Advanced Research,
Oak Ridge National Laboratory, Oak Ridge, TN 37831-6364

This paper describes our experience in using a multimedia project management and problem solving tool to support a multi-year Computer Productivity Initiative (CPI) project [1]. The purpose of the CPI project is to provide undergraduate students with a diverse set of skills required for the conception and development of solutions to large real world problems. In order to organize and develop a solution effectively, the students must work with a broad range of documents including video tape interviews of experts, presentation graphics for slide shows, computer drawn figures, charts and graphs, spreadsheets, as well as postscript of other forms of textual documentation. To organize and access the various media documents developed during the course of a CPI project, we are employing DHC (Decision based Hyper-multi-media CASE) tool developed to support the Decision Based Systems Development (DBSD) paradigm [2]. At the same time, DHC is used to guide and support the problem solving process the students employ to arrive at their solution. DBSD organizes the process of project management as well as the project document base around the decision making process. Our experience has demonstrated the value of a hyperlinked, multimedia project space in the development of large, group collaborative real world projects [3].

References

1. K.Maly, D.Ray, C. Wild, I. Levinstein, S. Olariu, M. Overstreet, N. Rao, T. Ireland, and G. Kantsios. Computer productivity initiative. *Proceedings of the 7th SEI Conference on Software Engineering Education*, pages 441–459, Jan 1994.
2. Chris Wild, Kurt Maly, and Lianfang Liu. Decision-based software development. *Journal of Software Maintenance*, 3(1):17–43, March 1991.
3. Chris Wild, Kurt J. Maly, Chenglin Zhang, Dennis E. Ray, Irwin B. Levinstein, Stephan Olariu, C. Michael Overstreet, Nageswara S. V. Rao, Daniela Rosca, Tammy Taylor, Tijen Ireland, and Deane Sibol. A decision-based hyper-multimedia case environment for the computer productivity initiative, a large undergraduate project. *Old Dominion University, Department of Computer Science, Technical Report*, (TR-93-25 (http://www.cs.odu.edu/WATERS/WATER-GS.html)), Nov. 1993.

Improving qualitative reasoning with an anchored computer simulation

SUSAN M. WILLIAMS

Learning Technology Center, Vanderbilt University, Box 45 GPC, Nashville, TN 37203 USA

Research in cognitive science has demonstrated that expert problem solvers have "deep" knowledge about the structure of their domain, i.e., they understand the relationships among variables in the domain and know the general principles that help to make predictions and solve problems. In contrast, novices remember problems in terms of easily observable surface features that may not be important to the problem's solution. Interacting with computer-based simulations can help novices move from knowledge representations that are based on the surface features of specific problems to representations that are based on the deep structure of general classes of problems. In the typical instructional approach, novices create a series of related problems by changing the values of variables in the simulation and observe how the corresponding outcome changes when the simulation is run.

There are, however, several reasons that novices may fail to develop more expert knowledge representations while using simulations: First, using a simulation requires some initial knowledge. If novices do not have this knowledge, they may be unable to use the simulation or to understand the feedback that it provides. Second, they may be unsystematic in exploring the problem space and may create a series of unrelated problems. Third, even when problems are related, novices may fail to compare successive executions of the simulation appropriately.

Anchored simulations can minimize these problems by situating learning in the context of a single problem over an extended period of time (CTGV, 1992). First, prior to using the simulation, students view a video that presents a complex mathematics problem as a story and work with their classmates and teacher to learn to solve this "anchoring" problem. In this way, they acquire basic knowledge of the problem and its solution that provides a foundation for using and understanding the simulation. Second, students cannot create unrelated problems, because an anchored simulation only enables creation of variants of the anchoring problem. Third, an anchored simulation creates an activity that causes students to automatically reflect on differences between problems.

The prototype simulation is anchored in a trip planning problem in which the main character purchases a boat and must decide if he has sufficient daylight and gas to drive the boat home. The student learns to solve this 16-step problem in class before using the simulation. Within the simulation, the student is challenged to a race by the main character and must make a single modification to an otherwise identical boat in order to win the race. The student then makes qualitative predictions about the race and confirms them quantitatively. When the simulation is run, the two boats race against each other, giving the student feedback on the predictions and calculations. The student is encouraged to undertake a systematic series of changes to the parameters affecting the boat's performance. Through this process, the student acquires a general model of trip-planning problems.

An in-school evaluation was conducted with 62 high- and average-achieving 6th-grade students who used the anchored simulation following classroom instruction in the anchoring problem. Use of the simulation helped students at both levels of ability improve their qualitative predictions about the outcome of trip-planning problems in the context of the anchoring problem and in other contexts (Williams, Bransford, Vye, Goldman, & Hmelo, 1993).

References

Cognition and Technology Group at Vanderbilt. (1992). The Jasper Experiment: An exploration of issues in learning and instructional design. In M. Hannafin & S. Hooper (Eds.), *Educational Technology Research and Development*, *40*, 65-80.

Williams, S.M., Bransford, J.D., Vye, N.J., Goldman, S.R., & Hmelo, C.E. (1993). *Using an anchored computer simulation to facilitate qualitative reasoning in mathematics*. Paper presented at the annual meeting of the American Educational Research Association, Atlanta, GA.

Hyper3D: Stereoscopic 3D Hypermedia

John E. Williamson
VRex, Inc., 8 Skyline Drive
Hawthorne, NY 10532

Hyper3D is a novel approach to interactive multimedia and hypertext/hypermedia presentations. Through the use of stereoscopic 3D LCD computer panels, the images are presented in stereoscopic depth with the ability to perform searches. This allows students to have both a better understanding of the material as well as increase attention and possibly retention. Prior to Hyper3D, there was no method of displaying and selecting stereoscopic 3D images which would allow both a large audience and assure high quality stereoscopic 3D images.

The long term goal of Hyper3D is to develop a complete history of stereoscopic photography from 1850 to the present day in digitized form. An estimated 8 million different commercial stereocard titles were released in their 60 year history in addition the many Hollywood movies produced during the 1950's. This does not include the many millions taken by amateurs or for office or educational uses. Two prolific amateur stereophotographers include President Eisenhower and silent film star Harold Lloyd.

Prior to Hyper3D, the majority of the public have not been able to see truly good stereoscopic 3D images. Subsequently, they are unaware of the long and varied history of stereophotography and more importantly the wealth of images represented in that format. Consequently, they have not been exposed to an important part of history in its original format. Using the flexibility of a hypermedia front end, the same images can be used to illustrate the history of photography, the development of photography as an art form, historical events, historical figures, architecture and any subject the user would wish to explore.

The ALPHA version of Hyper3D was designed on IBM 80286 with a Tektronix stereoscopic 3D monitor with the support of the USAF. The BETA version was developed on a IBM 80386 with a StereoGraphics stereoscopic 3D monitor. The final, commercial version is being developed on the VRex, micro-Pol system. This allows the conversion of nearly any active matrix LCD panel including stand alone monitors, laptops and project panels to be used. Both individual use on a laptop and projection to a large audience are possible.

Current chapters in Hyper3D include: Introduction to Visual Perception, The Evolution of Photography, How to Make Your Own Stereophotographs, Wonders of the World (both natural and manmade), Famous Personalities, Fashion Through the Ages, Weddings Through History, Religion, Sports, and Humor. Each of these will be released separately with links to each other allowing the user to add to their title collection through the same interface.

Each of these chapters may be searched on key phrases as well as be the image title, photographer, publisher and geographic location. Using the same database of images, several interfaces are used for various age groups and interests. These include very simple GUIs in a 3D Atlas style, allowing the user to click on a map location and subsequently showing an image from that location. These include both antique stereoscopic images from the 19th century, contemporary stereoscopic 3D images and computer generated reconstructions.

Other interfaces include a modified VCR interface allowing the user to page and fast forward through sequences of stereocards. Many sequenced stereocard sets where released in the 19th century that would explain an event (or even tell a story or joke) through sequence of stereocards, sometimes numbering in the hundreds of views.

References

Gibson, J. J., (1950). *The Perception of the Visual World,* Boston, Houghton Mifflin.
Julesz, B., (1971). *Foundations of Cyclopean Vision*, Chicago, University of Chicago Press.
Williamson, J. (1992). An Evaluation of Stereoscopic Computer Displays Using a Mental Rotation Paradigm. Unpublished masters thesis, College Station, TX.

CALL for German for Reading Knowledge

Peter Willmer
Dept. of Germanic Studies
The University of British Columbia
Vancouver, B.C.

Teaching foreign languages is often associated with a considerable degree of frustration: one seldom achieves a desired degree of fluency in a class, even after two or three years, let alone a single course. This certainly holds true for what one may term "standard language courses" - the kind through which one seeks to impart "total" knowledge of a language on a group of students. Considerable progress can, however, be made in the comparative short time of one academic year if one stresses solely one particular aspect of the language learning process, that of reading comprehension. At UBC we are currently producing a computer-based program that will enable students to gain competence in dealing with technical texts in German. With the aid of software that we are developing here, we hope to address the needs of students from various disciplines, assisting them in acquiring reading knowledge in German through working with texts from their particular field. We have also been careful to stress an approach through which students will master proper reading strategies so as to avoid the traditional pitfall of translation that is potentially destructive in the fostering of true reading ability.

Since we had emphasized reading strategies above all else, we were unable to find an authoring system that would meet all of our needs in the development of the software. We had envisioned a program through which the individual student would not only be tested on the content of a passage via short answer type questions, but also be given the opportunity to work with that particular passage in a way he or she desired, to mark the text in any of a number of different ways, identifying key elements as well as those which pose the most difficulty, or even (temporarily) deleting portions of the text. We have also developed the idea of answer screens with which students can, for example, compare their own summaries of a particular text with "standard" ones, that is to say, with our own. To integrate all of these features into our computer program, we were forced to link some form of word processing program to an normal authoring system. The result may be somewhat complex, but the program has already proved itself to be highly successful in our German for Reading Knowledge course. In the years to come, we hope to develop a new authoring system that will greatly simplify our present program.

Designing and Using Virtual Environments: The Advantage of Immersion.

W.WINN
College of Education and Human Interface Technology Laboratory, University of Washington, Seattle, WA., 98195, USA.

Although a variety of multimedia formats are now called "Virtual Reality" (Heim, 1993), the greatest innovation that the new VR technologies offer, and their greatest potential for educational applications, arise from those technologies that permit immersion. In an immersive environment, participants experience the illusion of being in another place in which they can move and look around in the same manner as in the real world.

An immersive environment has two advantages over non-immersive VR. First, it permits the participant what Clancey (1993) has called "first-person" experience of events. Much of formal education requires students to learn from descriptions of events and phenomena prepared by a third person. Immersive VR allows participants first-hand experiences. Second, interaction with a virtual envrionment can be achieved non-symbolically. Looking, pointing and manipulating objects in a virtual world are natural actions. It is not necessary to master a complex linguistic, mathematical or other symbol system before interacting with the virtual environment. It is therefore possible to master concepts before mastering symbols (Winn & Bricken, 1992), which is a great help to those who have difficulty thinking symbolically.

Because of these characteristics, the design of virtual environments for education has a different emphasis from "traditional" instructional design. Any knowledge or skill that are acquired from virtual environments are constructed by participants, not imparted didactically. This means that virtual world designers are concerned primarily with designing ways for participants to interact with worlds as easily, freely and imaginatively as they wish. The designer is therefore less concerned with presenting content so that it can be interpreted in one correct way, and is probably not concerned at all with the mastery of predetermined knowledge and skills described in performance objectives. Next, the designer is concerned with the appearance and behavior of objects, alone, in interaction with each other and with the participant. This is because the power of virtual environments stems not from their ability to simulate aspects of the real world but to make accessible to the senses and to interaction aspects of real and imaginary worlds that have hitherto not been represented. The designer is therefore concerned with such things as how to represent complex data sets, the inside of an atom, or the edge of the universe. Finally, the designer is concerned with how the participant may be guided -- not coerced -- along potentially fruitful paths. A virtual world without this kind of structure can be a disorienting place. This draws the designer's attention to the rules that govern the virtual world, and the extent to which they are enforced. For example, a world that obeys the laws of algebra rather than of physics could, if enforced, prevent participants from making mistakes when they factor algebraic expressions. Or they might be relaxed to allow mistakes from which participants will learn subsequently through "debugging". Varying the enforcement of rules governing virtual worlds provides a great variety of learning strategies for participants to employ.

References

Clancey, W.J. (1993) Situated action. A neuropsychological interpretation: Response to Vera and Simon. *Cognitive Science, 17*, 87-116.

Heim, M. (1993). *The metaphysics of virtual reality.* New York: Oxford University Press.

Winn, W.D., & Bricken, W. (1992). Designing virtual worlds for use in mathematics education The example of experiential algebra. *Educational Technology, 32*_(12), 12-19.

Design of An Inquiry Teaching Agenda Planner

LUNG-HSIANG WONG
School Of Applied Science, Nanyang Technological University, Nanyang Avenue, Singapore 2263

CHEE-KIT LOOI
Information Technology Institute, 71 Science Park Drive, Singapore 0511

This paper describes a project to study the feasibility of implementing the inquiry teaching method of [Collins & Stevens 1982] as tutoring software. In building the Tutoring Agenda Planner (TAP), we seek to integrate the two sets of architecture presented in [Collins & Stevens 1982] and [Peachey & McCalla 1986]. Collins and Stevens illustrated their theory, which was developed almost inductively by "observing" expert teachers. The theory which is meant to be domain-independent contains three parts: the goals and subgoals of teachers; the strategies used to realize different goals and subgoals; and the control structure for selecting and pursuing different goals and subgoals. Teachers typically pursue several subgoals simultaneously. Each goal has associated with a set of strategies for selecting cases, asking questions, and giving comments. In pursuing goals simultaneously, teachers maintain an agenda that allows them to allocate their time among the various goals efficiently. There are two top-level goals that teachers in inquiry dialogues pursue: (a) teaching students particular rules or theories; (b) teaching students how to derive rules or theories. There are several subgoals associated with each of these top-level goals. The dialogue control structure that the teacher uses to allocate time between different goals and subgoals is a crucial aspect for effective teaching. The control structure consists of four basic parts: (a) a set of strategies for selecting cases with respect to the top-level goals; (b) a student model; (c) an agenda; (d) a set of priority rules for adding goals and subgoals to the agenda. Given a set of top-level goals, the teacher selects cases that optimize the ability of the student to master those goals.

Peachey & McCalla introduced a course planner which is capable of planning global teaching strategies by using local information. The work reported here attempts to combine the planner portion of both sets of architecture. The integrated planner is complemented by an executor of plans with various teaching strategies and a domain-dependent natural language processor.

TAP is intended to serve as a tool which can be used to design the tutoring component of an intelligent tutoring system using an inquiry approach of teaching. The focus of the work is the planning process involved in conducting an inquiry dialogue. It is also an application of AI state-space planning approaches to the domain of planning teaching actions for an inquiry approach of learning.

We have written several sample inquiry dialogues in the domain of teaching the causes of rice-growing. With the assistance of a local gifted education specialist, we created different dialogues by postulating responses of students at different levels of competence. We have been able to generate some of these dialogues using our implementation of TAP using a pseudo-dialogue NLP component (see [Wong 1994] for a fuller account). TAP is intended to demonstrate the basic integrated architecture of GCP and ITP which enables combination of macro- and micro-planning techniques. Finally, some issues and problems concerning the implementation of inquiry teaching within the framework of a global course plan will be discussed.

References

Collins, A., and Stevens, A.L. [1982], "A Cognitive Theory for Inquiry Teaching", in P. Goodyear (Ed), Teaching Knowledge and Intelligent Tutoring, Norwood, N.J.: Ablex, 1991, pp. 203-230.

Peachey, D.R, and McCalla, G.I. [1986], "Using Planning Techniques in Intelligent Tutoring Systems", in Int. Journal of M-M Studies, Vol. 24, No. 1-6.

Wong, L.H., [1994], "An Agenda Planner for Inquiry Teaching ", internal report, School of Applied Science, Nanyang Technological University, 1994.

Developing A Framework for Delivering Technology-Enhanced Physics Instruction

JANICE E. J. WOODROW
Department of Mathematics and Science Education
University of British Columbia, Vancouver, B. C., V6T 1Z4

AUBRY FARENHOLTZ
D. W. Poppy Secondary School, Langley, B. C. V3A 4R1

GORDON SPANN
H. D. Stafford Secondary School, Langley, B. C. V3A 4J8

The Technology-Enhanced Physics Instruction (TEPI) Project was established to research instructional strategies for introductory Physics instruction which implement "state-of-the-art" technology. A recurring problem in Physics education is the inherent conceptual difficulty of the subject for many secondary students. This problem has been the focus of much research and debate. Emerging technology has made it possible to put into Physics classrooms, rich, problem solving environments and powerful, mind-extending tools that enable students to define and solve complex problems unencumbered by the necessity of first mastering complex quantitative skills. Content can be represented through a multiplicity of "languages" - verbal, mathematical, symbolic and graphic. But while the introduction of such technology affords the possibility of producing change and educational benefits, none of them can be assumed to become automatically realized only because of the technology's presence. The implementation of technology requires major adjustments in the traditional roles and instructional procedures of teachers. It takes careful preparation and planning to realize the educational benefits of technology and to make innovative things happen.

The Project has established a Macintosh-based, technology-enhanced environment in two Physics classrooms. Both classrooms are equipped with 8 student computers and 1 teacher computer, a printer, an active matrix projection panel, MBL equipment, and a multimedia center consisting of a TV monitor, a VCR, a laserdisc player and a CD-ROM player. By having two schools and two teachers involved in the study, the Project expects to develop a "thicker" description of effective instructional strategies as well as a "richer" context in which to conduct the study and a strong environment for collaboration.

The instructional focus of the Project is the use of computer-based simulations as both instructional and learning tools. By making Physics more visual (concrete) animated, and manipulative, simulations promise to motivate students, broaden student understanding of scientific principles and encourage more students to enter and remain in scientific studies. In addition, the Project is incorporating laserdiscs as both teaching and learning tools, multimedia technology for classroom presentations, MBL applications, and computer-based evaluation procedures. Plans and materials for implementing these resources are being developed and evaluated collaboratively by all three team members. The actual classroom teaching is being done by the two teachers as an integral part of their classroom instruction. The participating students view their use of technology, and that of their teachers, as part of the normal, year-long procedures of the classroom, not as some short-term novelty introduced for the purpose of research.

Results and Findings

1. A framework for integrating the many forms of technology-enhanced instruction and learning has been developed and successfully implemented.
2. Comprehensive Study Guides based upon this framework have been produced for eight Physics 11 units and eight Physics 12 units.
3. Student attitudes toward science and technology showed a significant increase.
4. The collaborative nature of the Project design has been extremely effective and perhaps essential in reaching the goals of the Project.
5. Both teachers have experienced a major shift in their instructional strategies.
6. The implementation of the technology engendered positive student collaboration, much increased on-task behaviour and increased student responsibility.

Natural Language Interface of Chinese on Multimedia and Education

Song Yunzian, Bai Peng, Jing Dong, Ilan Fanshi & Zhao Liang
Air Force College of Engineering, XI'AN, 7100`8, CHINA

The purpose of studying the natural language interface is to help the users to obtain the information from the system conveniently. Through many years' study, we have developed a regulated power-supply circuit trouble-diagnosis system (TEACH). This system can understand eleven English sentence patterns. Chinese is different from English, as the Chinese words are usually complexly combined and one word usually has several meanings, and it has no definite part of speech. There are no special dividing symbols between the Chinese words. All these make the natural language of Chinese difficult for a computer to understand. In practice we have put forward a set of rules based on the contained parts of speech in the multimedia education system. Thus the natural Chinese language interface has been developed.

The contained parts of speech can be divided into seven parts. Based on the above division, we can establish the dictionary bank with three segments: containing words, parts of speech and words meaning.

The application of the multimedia education system has achieved evident successes in class teaching and analogue training, and both the teaching efficiency and the training result have raised. The rules based on the contained parts of speech have been applied to the multimedia education system, which has broken down the conventions and has redivided the parts of speech. Thus the natural language understanding of the basic sentence patterns has been achieved and a feasible plan has been put forward in view of the specified branch of learning in natural language understanding. But whether this method is perfect or feasible to other domains has yet to be further proved in the future work.

References

Song Yunxian, Bai Peng: Knowledge Processing in Teaching Expert System for Trouble-Diagnosing, *Microelectronics & Computer*, Vol. 10 No. 3, 1990.

Bai Peng, Song Yunxian: The Natural Language Interface of Teaching Expert System, *Microelectronics & Computer*, Vol. 8 No. 8, 1991.

Song Yunxian, Han Fanshi, Zhao Liang: Applications of Natural Language Understanding of Chinese in ITS, 1993.

Learning Units and their Realisation Using New Media

H. ZAHRAN AND R. ZAIDAH
Faculty of Information Technology
Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia

Programmes of study are complex systems of teaching-learning activities which generally reflect current understandings and practices of a discipline, and anticipate future needs. Programmes, by this philosophy, must have built-in mechanisms and flexibility to react appropriately. However, many current programme designs are typically based on defining a set of courses. Courses are either subject-based or integrative, but nonetheless prescribed by the teacher. Such design frameworks suffer several shortcomings, for example (1) it is committed to a didactic, teacher-led mode, (2) the aggregation of educational material into courses and that they exist only in such a form hinders any significant form of individualised or negotiated learning, and (3) a course in practice is too large to be adapted quickly to advances in the discipline, changing market needs or social relevance.

Learning Units

Our approach to overcoming these is to uncouple the subject matter from their aggregation into meaningful courses, i.e. devise a high degree of independence between educational material and courses that utilise them. The creation and upkeep of such material are then not driven by course requirements but by the current body of knowledge. We call such educational resources "Learning Units" (LU). Such units may be broadly categorised as one of: (1) *Knowledge Units* (KU), with emphasis on the exposition of theories, abstractions and facts of the subject matter, (2) *Activity Units* (AU), with emphasis on know-how or skills acquisition, and are developed through active learning situations such as games or laboratories, and (3) *Experiential Units* (EU), with emphasis on real life application of acquired knowledge and skills, and are the opportunities created via project work or industry residency.

LUs are like *tangram* pieces that may be creatively juxtaposed later on to form interesting courses. They should be combined so that the composite subject matter forms a coherent body of topics. This approach provides greater flexibility than traditional course modules which tend to be of a fixed size and purpose. Different teaching-learning approaches can be supported, including problem-based/multi-disciplinary approach, learner managed learning, negotiated learning (Stephenson, J. & Laycock, M.,1993), and even traditional prescribed course delivery.

Global Digital Repository

Digital media integration, computers and communication systems define a *new media* to store and organise information for instant access, correlation, and assembly. Our basic strategy is to develop LUs on the new media, using for example the large hypermedia system, Hyper-G (Kappe, F. & Maurer, H.,1993). This will not only facilitate their maintenance, but also provide significant opportunities to realise new teaching-learning paradigms and innovative modes, more suitable perhaps to the Nintendo-generation. We seek international participation as this would help produce up-to-date repositories that can be made accessible to areas not having resident expertise. Such distribution and globalisation would therefore support true international cooperation in education.

References

Kappe, F., & Maurer (1993). H. Hyper-G . A Large Universal Hypermedia System and Some Spin-Offs. *SIGRAPH*, May.

Stephenson, J., & Laycock, M, (editors) (1993). Using Learning Contracts in Higher Education. *Kogan Page Ltd.*.

DEMONSTRATIONS/POSTERS

Object-Oriented Data Modeling in Hypermedia: A New Data Model

K. ANDREWS, H. MAURER, N. SCHERBAKOV
Institute for Information Processing and Computer Supported New Media(IICM), Graz University of Technology, A-8010 Graz, Austria.

Although the object-oriented paradigm is well suited for the modeling of self-contained independent objects, it is not suited for modeling relations (or static links) between abstract data objects. At the same time, the concept of computer-navigable links is a main part of the hypermedia paradigm. In contrast to multimedia, where the object-oriented paradigm plays a leading role, this "confrontation" considerably reduces the application of object-oriented methods in hypermedia.

The HM data model incorporates the well-known principles of object-oriented data modeling for actual management of large-scale, multi-user hypermedia databases. The model is based on the notion of abstract hypermedia data objects called S-collections. Computer navigable links are encapsulated within a particular S-collection and are also bound between S-collections. Thus, a hypermedia database is nothing but a set of *S-collections*. Each S-collection has a unique identifier and a so-called *content*. The content can be seen as a primitive node attached to an S-collection. The content (text, pictures, audio, video-clips, etc.) is visualised when the S-collection is accessed in some way.

An S-collection encapsulates a particular navigable structure . The navigable structure is a set of other S-collections (called *members* henceforth) related by a number of computer-navigable links. Note that we connect S-collections but not primitive nodes. Note also that the links are encapsulated within a particular S-collection. In other words, links may be defined only between members of a certain S-collection , and in this sense, links belong to such S-collection, but they do not belong to a hypermedia database or to members that are related by means of links. An S-collection devoid of such an internal structure is called a primitive S-collection.

All S-collections can respond to the message "*access*". It implies executing the S-collection's content (i.e., presenting some text, picture, audio, video clips etc.). Typically, a chunk of hypermedia information associated with the current S-collection is visualised on the user screen, but any kind of action can happen in response to the message "*access*" if the corresponding method has been overridden. Complex S-collections without content forward the message to their head.

Link following within the HM Data Model is simply a form of message passing. At any particular moment in time, the user can navigate only through a single, specific S-collection called the *current container*. Only members of the current container can receive messages during navigation. A concrete member of the current container is the *current member* for each particular navigational step. More precisely, the member that most recently received the message "*access*" is the current member. Only members related (linked) to the current member can be accessed (can receive the message "*access*") in the next step of navigation.

Since links are encapsulated within an S-collection, they become available (or become activated) for navigation only if the S-collection has been "entered". Such an *entering* of an S-collection to activate the encapsulated links will be called the "*zoom in*" operation. In analogy, the operation "*zoom out*" is the complement to the most recent "*zoom in*" action. More precisely, the operation "*zoom out*" recovers current container and current member to the state they had just before the previous "*zoom in*" operation. Extending the functionality of "zoom out" to give access to any S-collection of which the current collection is a member is provided by the operation "zoom up". Generally, the "zoom in", "zoom up" and "zoom out" operations provide users with the possibility to navigate in direction orthogonal to the conventional plane of link-based browsing. Thus, we can say that the HM data model supports an additional dimension of browsing hypermedia databases.

The HM Data Model has been implemented as a prototype system called HM Card, The system, its documentation, and a number of articles are available via anonymous ftp from "iicm.tu-graz.ac.at" in directory "pub\hmcard".

Software to Involve the Instructor in Creating Multimedia Listening Comprehension Lessons

STEWART ARNEIL
Department of the Language Centre, University of Victoria
Victoria B.C., V8W 3P4 ,Canada

The Goals and Challenges

Our research supported the sense of some students, faculty and staff of a lack of student material requiring comprehension of, and engagement with, spoken language. We decided to develop such material based as much as possible on principles used in reading comprehension exercises. Development would be computer-assisted - exploiting instructor's teaching and content expertise and interests, and minimizing their design, development and detail work. Self-administered listening lessons require the use of mechanical devices, and the control of those devices in response to student input calls for a computer. We had to accept limits on the degree of control of audio/video presentation and on accepting and evaluating natural language input, especially sound.

The Solutions: Ideal and Actual

Ideally, we want to allow authors to apply their instructional goals easily to a variety of source media (audio/video on disc or tape, text, graphics, digitized audio/video) by accepting a variety of input from students (typing, clicking, utterances), and using a variety of output media (computer/video monitor, speakers, printer).

Currently, our program accepts videodisc, CD audio, text and graphics. We intend to add animation and digitized sound. We do not yet accept oral input from the student because of its technical demands and our inability to evaluate it. The program corrects only short pieces of text (a few words). Testing indicated that tape players were not mechanically practical, and that pressing CDs can be a viable alternative.

We hoped to be able to divide the process of developing a lesson into modules, and provide a computer program for each module. Ideally, the modules would reflect different skill sets (such as planning versus data entry), and should reward, but not require, advance planning of material and of the development process. Each module's program should make suggestions at each step, but allow the author to override whenever possible.

After research, we divided the development process into Planning, Detailing and Creating modules. The Planning module of the computer program helps the author (likely a professor) decide on the overall structure of a lesson, the number and type of instructional goals, and of activities (such as multiple choice) for each goal, drawn from a total of about twenty. The Detailing module helps the author decide for each activity which elements (clips, introduction, hints, feedback etc.) to include, and to fill in as many details as are desired. The Creating module prompts the author (likely a teaching assistant) step-by-step to fill in the required information for each element of each activity. It then adds the activity to the program the student sees.

The program the student sees allows him or her to follow the path set by the author, move to any activity in the lesson, repeat activities that were answered unsatisfactorily, or search other lessons for related activities.

Still to Go (or Making Multi Media Marginally More Manageable)

Transfer source audio and video from tape to disc, and simplify the transfer process.
Solve technical and instructional problems introduced by recording student utterances.
Improve listings of instructional goals and mapping of instructional goals to activities.
Complete full-scale testing of planning, detailing and creating programs and resulting student material.
Minimize features that cause authors to construe formalizing lessons as mere computer programming.

References

Hendricks, Harold. (1991). Models of Interactive Videodisc Development. In *Proceedings of the Second Annual International Conference on Foreign Language Education and Technology* .

Richmond, Ian M. (Unpublished). Designing Foreign-Language Reading Software for Both Students and Teachers. Manuscript obtained privately.

LECTURE: A Macintosh Multimedia Authoring System and Self-instructional Student Video Program

R. BALAS
Department of Foreign Languages and Literatures
Western Washington University, Bellingham WA 98225 USA

D. DENTON, 2ND AUTHOR
Media Services
Western Washington University, Bellingham WA 98225 USA

Lecture was created to provide a self-instructional vehicle to the student for the total comprehension and mastery of authentic language video in the CAV laserdisc format. Exploiting video of native speakers directing themselves to other native speakers creates too many classroom obstacles to the student in terms of comprehension and feedback problems and to the instructor in terms of time restraints to permit adequate utilization and mastery of such video material in a normal classroom situation. Our first objective was to provide the student with the means to raise his/her aural understanding from 95 percent to 100 percent of the spoken target language. As the program developed, we saw the need for variety and breadth in the self-instructional program in order to enable the student to explore and practice the language being comprehended through problem-solving and a hyperenvironment which would permit a variety of activities built around selected video segments.

The project required both an overall student program design containing varied and demanding student input activities and an authoring program which facilitated the easy development of such activities. The result of our effort was the creation of the Lecture Authoring Program which permits the exploitation of CAV laserdiscs through simple typing and "point and click" lesson production. Lecture now links texts to video frame numbers and to specially designed internal dictionaries which in turn link to other texts, graphics, QuickTime video, CD-ROM, and sound files. The program also contains reinforcement exercises (multiple choice, fill-in-the-blank, and question/answer exercises with error feedback), a composition creation activity, a stack editing feature, the ability to convert a student program to another target language, and the ability to build activities on a second laserdisc sound track when available.

In the demonstration we will exhibit the student self-instructional program and discuss the pedagogical rationale behind the program's structure, its varied activities, and the emphasis on student input. We will show the flexibility inherent in the self-instructional program when used in conjunction with video and also discuss how the program can also quickly create classroom presentations which in turn feedback to the self-study program. Although the program may be used and converted to a variety of languages, we shall use examples from the intermediate to advanced ESL programs based upon Welcome to Bellingham, a laserdisc created at Western Washington University. I may also include excerpts from the beginning Italian BBC program, Buongiorno, Italia, and/or France-TV Magazine, French in Action and Destinos.

Next we will give a survey of the Lecture Authoring Program and explain its case of operation in creating text and video control exercises, multiple choice, fill-in-the-blank, question and answer, and composition creation. We will demonstrate how the editing capacity of the program permits all aspects of the student program to be developed, edited, and amended.

Multimedia Technology in the Calculus Classroom

MAUREEN BARDWELL AND ANNE PASQUINO
Departments of Mathematics and Computer Science
Westfield State College, Westfield, MA. 01086, USA

Over the last ten years there has been an ever increasing focus within the mathematics community to examine and modify the traditional college calculus sequence. It has become apparent that the "chalk and talk" presentation style is not successful with many of today's students and that the technology that is now readily available to calculus students forces their teachers to reexamine what material they are teaching to their classes as well as how the presentation should take place. Most of the National Science Foundation funded Calculus Reform Projects strongly encourage the incorporation of technology into the mathematics classroom. Typically, the technology that is used is a graphing calculator or a PC with a graphing software package. In this poster session, we will demonstrate some multimedia applications that are used for presentation and tutorials in Calculus I and II at Westfield State College.

Westfield State College has gone through a natural progression in its reform of the Calculus sequence. Initial change in the course was introduced by assigning occasional computer labs with a graphing package on a PC. At present, we have just concluded our fifth year of teaching a Reform Calculus course. Major components of the course now include assigned computer labs, the requirement of a TI82 graphing calculator, group projects, and the use of a Reform Calculus text.

The use of multimedia to present application problems in calculus is a natural extension of the use of technology within the calculus courses. The development of calculus was initially motivated by physical problems, and today's student is more clearly able to appreciate this if the situation being modelled can actually be demonstrated at the time of study. Video is a natural medium for presentation of physical problems. The use of the authoring package Toolbook allowed development of calculus lessons where difficult points could be reviewed and expanded upon by the learner; it also gave us the opportunity to present the visual and auditory reinforcement which is helpful to today's student.

References

Dick, & Patton. (1992). *Calculus*. Boston, MA: PWS Kent.
Hughes-Hallett, Gleason, et. al. (1993). *Calculus*. New York, NY: Wiley Publishers.

Creating a Multimedia Infrastructure for Faculty Development

ELLEN YU BORKOWSKI*,
NOEL ANDERSON†, WILLIAM MAKI††, VAL TARESKI†, AND RICHARD HANSON**
†Department of Electrical Engineering, ††Department of Psychology
**Multimedia Project, **Vice President for Academic Affairs*
North Dakota State University, Fargo, North Dakota 58105 USA

North Dakota State University (NDSU) is establishing an aggressive, university-wide approach to incorporating multimedia into the classroom. NDSU has been working on building an infrastructure of support and resources for faculty development of multimedia in the form of personnel, equipment, training and support for information sharing.

A multimedia project was funded through an internal funding process run by the University's Planning, Priorities & Resources Committee (PPRC). The project was formed through the combination of ideas from several of the proposals submitted that were related to multimedia. The faculty involved in the funded multimedia proposals have formed a campus Multimedia Coordinating Committee (MCC).

Personnel for supporting the multimedia project came in the form of the MCC, a full-time multimedia specialist and student assistants. The MCC has defined ambitious goals for supporting multimedia on campus. The multimedia specialist supports the component parts of the multimedia project. Her responsibilities include taking care of the equipment, conducting training, working with faculty and supervising the student assistants. The student assistants were trained on the authoring software taught to faculty and provide technical support (telephone and on-line) to faculty. Other services, such as slide scanning and CD-ROM mastering, are also handled by the student assistants.

Understanding that many of the faculty who wanted to get involved with multimedia did not have the funds to purchase high end computers for their office, the multimedia project purchased authoring stations which were made available to faculty. To support both platforms, a Macintosh Quadra 800 and an IBM Ultimedia Model 77 were purchased. Both have video and audio capture and editing ability. A 35mm slide scanner was purchased and is supported on both the Mac and DOS/Windows platforms. To handle s rage and access to the growing amount of information associated with multimedia, a CD-ROM mastering unit was purchased. This also is supported on both the Mac and DOS/Windows platforms. A third computer (DOS/Windows based) was purchased to support slide scanning and CD-ROM mastering.

Another "form" of equipment available to faculty is a multimedia classroom. This is a 180-seat lecture hall which is equipped with a Mac and an IBM computer along with several audio/visual devices. All video is projected through a ceiling-mounted SONY projector. Audio is played out of a stereo sound system.

Training is conducted by the multimedia specialist. Introductory and advanced workshops on selected authoring tools were offered. Both Macintosh and PC classes were conducted.

Support for information sharing were offered in two mediums. A multimedia user group (MMUG) was formed which met monthly to hear presentations and have discussions about methods and software and how to apply it in the classroom. Electronic support was also provided through the creation of a local newsgroup. Announcements of upcoming meetings, software tips, and conference announcements from Internet were posted to this newsgroup. It also supported on-line discussions & questions from faculty.

Many lessons have been learned about slide scanning, CD-ROM Mastering, and courseware development. They include: acquire wisely, be realistic, start small, establish a scheduling priority for appropriate use of multimedia classrooms and equipment, and be aware of copyright issues.

For successful implementation of campus-wide use of multimedia campus-wide, it is crucial that a foundation of support be built. Providing technical support through training and hiring of support personnel, providing funding to purchase the required hardware and software and providing the means for fostering partnerships through user groups and on-line discussion builds the necessary foundation that faculty can "stand on" to start developing multimedia materials.

The Shaping of Knowledge: Aesthetics in Designing and Engagement in Educational Multimedia

J. Blatter and A. Breuleux
McGill University; Montreal, Quebec, Canada H3A 1Y2. Internet: ed13@musica.mcgill.ca

The development of *Collector's Box* a HyperCard stack to be used as an off-site natural history museum enhancement led us to consider: a) the design process rather than simply the product, and b) explore the aesthetic point of view on design, i.e., the epistemological issues of interaction between designer, object, and viewer.

Aesthetics, a branch of philosophy, provides reflections on the process of creating objects and events that engage users. Creating computer-based multimedia environments may be considered in a similar way. And, in effect, issues of engagement and motivation are being included in current research on educational multimedia. Aesthetic issues, far from being decorative or secondary to the instructional goals (Levin, 1987; Rieber, Boyce and Assad, 1990), form an inherent part of design, furnishing the basis for its motivation and evaluation. They allow the designer to mentally juggle the instructional and multimedia design components in order to create an organic, meaningful whole. This analysis draws on the interaction between cognition, motivation, and aesthetics (Berleant, 1991; Dufrenne, 1987; Polanyi,1975), and the rhetorical dimensions of art and design (Buchanan,1989; Eco, 1979).

Similarly, the *role of the user* (to paraphrase Eco, 1979) in this aesthetic process, is one of involvement, interpretation, and "imaginative integration" (Polanyi, 1975) in order to construct coherence, meaning and knowledge. This provides the basis for investigating genuine engagement between multimedia and user.

We will be illustrating aesthetic considerations that occurred in the design of four features of the *Collector's Box*, presenting the most recent version of the stack in progress, along with digital "sketchbooks" showing the aesthetic, problem solving process of idea, action, judgment, and re-action. The four features are: 1. stack metaphor; 2. "field tools"; 3.user "field notebook"; 4. networking. Briefly, the *Collector's Box* is a multimedia museum enhancement for Grade 6-12 that presents fundamental concepts and procedures of identification and classification used by natural historians, and seeks to positively affect attitudes toward natural history. Basically, the *Box* encourages students to think as natural historians, and stimulates the students' interest to explore the "real world".

Stack development began with metaphors establishing the relationship between the student and the *Box*. Metaphors furnished the visual inspiration and structure needed to dir ect, objectify, and unify domain and instructional information, and resulted in stack structure. Meanings associated with the metaphor of a collectors' box generated ideas for other instructional features: "field tools", including binoculars, magnifiers, video, etc., support multiple representations and enhancement of observation skills; user's"field notebook", including graphic and text applications, encourages user cognitive strategies, and gives the user symbolic and functional screen space; networking applications linked to the *Collector's Box* provide a "real world" overlay to the metaphorical context.

References

Berleant, A. (1991). Art and Engagement. Philadelphia: Temple University Press.

Buchanen, R. (1989). Declaration by design: Rhetoric, argument, and demonstration in design practice. In V. Margolin, (Ed.), Design discourse. Chicago:University of Chicago Press.

Dufrenne, M. In the Presence of the Sensuous: Essays in Aesthetics. (1987). Atlantic Highlands, NJ: Humanities Press International, Inc.

Eco, U. (1979). The Role of the Reader. Indiana: Indiana University Press.

Levin, J.R. (1981). On the functions of Pictures in Prose. In F. J. Pirossolo & M.C. Wittrock, eds. Neuropsychological and Cognitive Processes in Reading. San Diego: Academic Press.

Polanyi, M., and Prosch, H. (1975). Meaning. Chicago: Chicago University Press.

Rieber, L.P., Boyce, M.J., & Assad, C. (1990). The effects of computer animation on adult learning and retrieval tasks. Journal of Computer-based Instruction, 17, 46-52.

Intelligent Strategies For The Presentation And Interpretation Of Video In Intelligent Tutoring Systems.

S. BUTLER AND A. P. PARKES
Computing Department
SECAMS, Lancaster University, Lancaster, Lancashire, LA1 4YR, U.K.

Multimedia systems can now have cheap video, and Multimedia Authoring Systems are being used for creating Intelligent Tutoring Systems (ITS). Thus interactive film/video is now available as a resource for ITS designers. Film however must be used carefully because film clips take on new meanings when juxtaposed with other clips. D. Kuleshov in 1920 conducted an experiment (from Pudovkin (1958)) which highlights several aspects of film clip juxtaposition. Visualise five clips of film...

1. A young man walks from left to right. 2. A woman walks from right to left. 3. They meet and shake hands. The young man points. 4. A large white building is shown, with a broad flight of steps. 5. The two ascend the steps.

"The pieces, separately shot, were assembled in the order given and projected upon the screen. The spectator was presented with the pieces thus joined as one clear, uninterrupted action; a meeting of two young people, an invitation to a nearby house, and an entry into it. Every single piece, however, had been shot in a different place; for example, the young man near the G.U.M. building, the woman near Gogol's monument, the handshake near the Bolshoi Teatr, the white house came out of an American picture (it was, in fact, The White House), and the ascent of the steps was made at St. Saviour's Cathedral What happened as a result? **Though the shooting had been done in varied locations the spectator perceived the scene as a whole.**"

It is investigation of film phenomena like this that leads me to formal rules for film editing.

The strategies (from the title) consist of a film structure, an event structure and a set of rules for translating the event structure into the film structure. This approach was first described by Carroll (1980), however I have taken his embryonic film grammar and extended it with particular relevance to film being presented as part of an ITS or any Multimedia Application. I feel Carroll experienced problems with his film grammar because he scoped it too widely. By narrowing a grammar's application area I feel it becomes more relevant. A fragment is shown below.

1. Scene	==>	Scene*
2. Scene	==>	Establishing Shot + Detail Shot*
3. Scene	==>	Detail Shot* + Revealing Shot
4. Establishing Shot	==>	Extreme Long Shot | Long Shot
5. Revealing Shot	==>	Extreme Long Shot | Long Shot
6. Detail Shot	==>	Detail Shot + Scene + Detail Shot

(* meaning one or more , + meaning concatenation, and | meaning exclusive-or)

The problem with such a notation for expressing the structure of film is that the structure is dependent on what you want to communicate. Thus, on the whole, rule 2 would be used for scene structure, but rule 3 used when we want to bring the viewer to a particular point then reveal some extra information which shows them the actual conclusion. Such strategic decisions would be made by the ITS. The ultimate aim is to amalgamate this grammar with a knowledge base describing the content of some film (see Parkes (1989)) into an ITS which dynamically edits the film for presenting a given concept and parses user input in the same manner. Research is currently underway at Lancaster into this problem.

References

Carroll, J. (1980) *Toward a Structural Psychology Of Cinema*. The Hague: Mouton
Eisenstein, S. M. (1942) *Film Sense*. New York: Harcourt, Brace World.
Parkes, A. P. (1989) *An Artificial Intelligence Approach to the Conceptual Description Of Videodisk Images*. Ph.D. Thesis. Lancaster University.
Pudovkin, V. I. (1958) *Film Technique*. London: Vision, Mayflower.

Computer Use with Adult Students: Addressing Multiple Learning Styles

PATRICK O. CHAMBERS
Department of Educational Administration and Foundations
Western Washington University, Bellingham, Washington, 98225-9087, USA
patc@henson.cc.wwu.edu

All individuals do not process information in the same ways. For some, a verbal description or a lecture can be a very meaningful form of learning. Others process visual information much more readily. Still others need the physical act of writing something down to fully process it. And most individuals who have been successful in the educational system have learned to gain information from printed media. The degree to which meeting of the individual's learning styles has selected thosewho have succeeded is only now beginning to be more clearly understood. Recognition of various learning disabilities is one example of this.

Some individuals who work with adults have come to recognize that an adult's learning style may play a very significant role in learning, and that it may have had an effect on past educational success. It is not easy to always meet these various learning style needs. And when those working with adults attempt to incorporate computer-assisted instruction into the learning program, many adult students, who may be more technophobic to begin with, do not always find a good match between the approach of the software and their individual learning styles. The difficulty in learning continues. But designing software that accommodates multiple learning styles, while more work, may pay significant benefits in meeting the needs of this broader audience.

For purposes here, the following learning styles will be considered, with the recognition that even this list is not complete. Print-based learning is the most commonly used method of acquiring information and may be the most *acquired* learning style; one that individuals have adapted to in order to succeed in the established educational system. Individuals who cannot adapt to this style often find educational failure, both as children and adults. Auditory learners are those who need to hear something to fully process it. In lectures they may sit with their eyes closed to focus on the auditory stimulus. Visual learners may focus more on what is shown than what is said. For them, a picture may be worth a thousand words. Graphs and charts may also be quite helpful in retaining information. These individuals can frequently *see* the information in their mind. Interactive learners may need to respond to the information to fully incorporate it. For example, they may need to repeat the information aloud or respond verbally to it. Kinesthetic learners often need to be *doing* something (or moving) while learning. For some this may take place by physically writing the information. For others it may mean doing something while listening—like knitting or doodling. And, it is possible that another learning style may involve the keying of information to symbols. Few people have only one learning style, but commonly one is prominent.

The Demonstration Programs

A short diagnostic element at the beginning of a program determines some characteristics of an individual's learning style and in turn accesses a matrix of learning elements designed to fit that style. The programs contain modules appropriate to different styles but covering the same material. Students use the software modules covering the material in the most appropriate manner. An added benefit of this type of programming is that having available the additional modes for use after a student has viewed material in a primary mode can be used to reinforce the learning in the activity.

This demonstration will utilize simple HyperCard™ programs designed to do a very simple diagnosis of the learning style of the user and then demonstrate some strategies for meeting those needs. One of these programs was created to show graduate students in adult education the nature of some adult learning styles and how given material can be presented to more closely meet those needs. These programs cover print-based, visual, kinesthetic, interactive and aural based elements. Some symbols have also been incorporated. An additional goal in these programs was to encourage practitioners to develop their own simple programs to meet specific learning goals.

Learning Research Skills: a HyperCard Lesson

MAY YING CHAU

Sciences & Technology Department
Colorado State University Libraries
Fort Collins, CO 80523, USA

Information retrieval skills are essential for successful research and the achievement of sound scholarship. Students should be taught efficient research methods as early as possible so that they can develop critical thinking skills. In order to help students acquire research capability, Colorado State University Libraries offers scheduled classes each semester to teach students efficient research methods. Although these classes are highly successful, student enrollment is limited. An alternative for students who cannot enroll in the scheduled library classes is, in the opinion of the author, crucial.

In recent years, multimedia technology has been implemented as a learning aid in the classroom. Multimedia is a system that combines text, graphics and sound to present training and information (Dahmer 1993) and is an excellent tool for a non-traditional learning environment. Students can engage in interactive dialog with the computer and choose various lessons from the menu provided. Although this HyperCard program is not exactly a multimedia courseware, the author has incorporated text, graphics and animation into the content. Menus allowing student to choose their own learning options provide interactivity.

The stated objectives of this HyperCard program are:

* familiarize students with the physical layout and collection arrangement in the Colorado State University Libraries.
* teach students to use SilverPlatter CD products.
* help students to define and identify different types of reference sources.
* guide students through the intricacies of using the CARL system.

Since computer learning programs provide students with an individualized learning environment, students from diverse groups who may have different learning styles (Wayman 1984) and language barriers would benefit from this HyerCard program.

Future Planning

This HyperCard program is a preliminary study for future multimedia projects. The future goals include:

* incorporate sound - music and speech (in different languages)
* create full-color versions
* create 2 versions - Macintosh (HyperCard) and IBM (ToolBook)
* include reference sources in different disciplines
* demonstrate research methods - how to narrow down the topic and verify information

Reference

Dahmer, Bart. (1993) When Technology Connect. *Training & Development*, 47(1), 46-55.

Wayman, Sally G.(1984) The International Student in the Academic Library. *The Journal of Academic Librarianship*, 9(6), 336-341.

UTLearn: Interactive, Online Instruction at the University of Toronto Library

M. CLINTON and S. KASZUBA
Information Technology Services, University of Toronto Library, Toronto, M5S 1A5, Canada

How to Reach Large Library User Populations with Instruction

The University of Toronto has three large campuses, over 60,000 students, and 49 libraries. In 1992, the University of Toronto Library (UTL) introduced a new online information system, called UTLink which contains the 7-million-item UTL Catalogue, online periodical indexes, a campus and library gopher, and other programs. Many students and others using UTLink and other online library services, access these remotely from their home or office by modems or through networks.

Instructing such a large and diverse library user population, which includes undergraduates, graduates, and part-time students is a challenge. The Library continues to use classroom-based instruction and other methods to teach information searching skills and strategies. But to meet the challenge of numbers in a more comprehensive way, the Library has developed an online interactive learning module called UTLearn.

UTLearn Offers Interactive, Online Instruction

UTLearn consists of two programs: MedLearn, which instructs users on how to access and search the locally installed Medline databases; and Teach Yourself, which contains information on how to search catalogues and databases, research and write essays, and find information in libraries. UTLearn contains more than 500 screens of information, is self-paced, and allows users to choose the level of detail they want to view. It contains both concept-based and procedural units. The premise of the concept-based units is that procedures on different systems vary but that general search principles remain similar on many systems. The procedural instruction units deal with the specifics of searching the Library's catalogue, which uses the Data Research Associates (DRA) interface, and the locally installed CD-Plus Medline.

The UTLearn project grew out of the Library's previous CBL programs in HyperCard and studies of users' learning methods and preferences on University of Toronto Library system. UTLearn offers a cost-effective way to reach more users than possible by other instructional means.

UTLearn: Successes and Limitations

- approach of choice for many learners
- gives identical approach in all locations
- available whether or not staff is available
- reaches a large audience of remote learners
- eliminates interpersonal bias (gender, race, age)
- gives no opportunity for real-time dialog
- is relatively easy to do and update

References

Cherry, J M., & Clinton, M. (1992). OPACs at five Ontario Universities: a profile of users and user satisfaction. *Canadian Library Journal*, 49, 123-133.

Cherry, J M., & Clinton, M. (1991). An experimental investigation of two types of instruction for OPAC users. *Canadian Journal of Information Science*, 16, 2-22.

Clinton, M., & Kaszuba, S. (1991). A library instructional program for teaching computer-based information-seeking skills and strategies to undergraduate students at the University of Toronto. In *Proceedings of the 12th E.C.O.O. and the 8th I.C.T.E. Joint Conference*. Toronto, May 3-10, 20-22.

An Architecture for Interactive Hypermedia Training Systems

DORIS F. DE ARAGON, AND LAURA MARIA COUTINHO LOPES
CELTEC
Av. Epitácio Pessoa, 874 - Ipanema - Rio de Janeiro - Brasil - 22471-000

MÔNICA ARAGON
Departamento de Inteligência Artificial - ILTC
Rua Almte. Teffé, 637 - Niteroi - Rio de Janeiro - Brasil - 24030-080

We developed an Architecture for Interactive Hypermedia Training Systems based on our experience in constructing Knowledge Based Systems, Hypermedia interfaces and Computer Based Training. To test these ideas we have prepared a prototype system to train people to repair a car.

The architecture consists of 4 modules: an interactive hypermedia interface, a knowledge base, an inference engine and a pedagogical strategy including evaluations.

The hyperdocument permits the user to choose among the options and thus create his own training.

The development of the architecture considered the following aspects:

- The advantages of using a Multimedia interface (texts + figures + images + videos and so on) which provides motivation and makes possible the use of both sight and hearing senses. (Researches in training shows that learning is more significant when students' senses work together.)
- The interactive power of hypertexts giving the user the freedom of making his own training.
- The necessity of a pedagogical strategy with evaluations orientating the student and completing the learning cycle: motivation, perception, understanding, assimilation and the transfer of knowledge being exploited.

This Interactive Hypermedia Training System generally speaking, also stimulates the development of a critical sense once a constant interaction with user is needed. Aspects such as agility and organization of thoughts, as well as formulation of new ideas are also developed by trainees. When using this system Multimedia provides the motivation and the use of all senses helping to improve perception.

Conclusion

Our experience in constructing such systems has shown that putting together technologies on Education, Multimedia and Artificial Intelligence improves Computer Based Training Systems substantially.

References

Aragon, D.F et al. "A Learning System using Hypermedia." 12th E.C.O.O and 8th ICTE Joint Conference, Canada, 1991.

Aragon, M & Evsuboff, A. "A Multimedia interface for trainig systems", Technical Report, ILTC, 1993.

O'Neal, F.. "A Model for Computer - Based Training". AI Expert, december, 1992.

Wood head, N.."Hypertext and Hypermedia Theory and Applications". Addison Wesley Public, 1991.

Computer supported co-operative problem handling

D. J. DETOMBE

Faculty of Policy Analyses, System Engineering & Management
Delft University of Technology, P. O. Box 5015, 2600 GA Delft, The Netherlands

The method COMPRAM supports the process of co-operative problem analysing of complex interdisciplinary societal problems. Interdisciplinary problems are problems which are often undefined, on which it is hard to find a 'solution', which has a great impact on society, and in which many people are involved. Knowledge and data of these problems are only partly available. An example of a complex interdisciplinary societal problem is the Aids problem, education in general, or building a world wide computer infrastructure. The method is specially developed for those problems, which can be analysed in causal relations. It can be applied to policy problems, but also for learning creative co-operative problem solving in education.

The knowledge of a complex interdisciplinary societal problem is, as far as it is available, known by experts of the different disciplines. Each expert has a good and detailed view of the problem on his or her domain, however often a vague view of the whole problem. In order to get an overall view of the problem, what phenomena are involved and how they are related, the knowledge of the different domains should be combined, and the white spots should be filled. In order to combine the knowledge the experts should come in contact with each other, discuss and exchange their domain knowledge in order to construct in a mutual effort the conceptual model of the problem.
However communication in a multi-disciplinary team is difficult. Except from the normal problems in group-interaction such as hidden agenda's, group think, and blind spots, there is the difference in professionally habits, language and methodology.

COMPRAM is an interactive method that supports the information exchange of a team of multi-disciplinary experts in their attempt to analyse complex interdisciplinary societal problems. The method is based on ideas coming from cognitive psychology, computer science, theories about group-processes, using system dynamic modelling and group decision support tools. The method can be regarded as a framework which includes several information retrieval methods from social science in combination with computer tools to support the exchange. In order to cope with the different professional background of the team, the method emphasizes the defining of concepts and the explanation of the theoretical ideas. A six layer model is used to express the model. In this model the problem is next to words, also expressed in graphical models, such as a semantic model, a causal model and a simulation model. The six layers together express the conceptual model of the problem. These different ways of expressing the problem makes it easier to understand the way the phenomena are related with each other and create. a kind of mutual language.

Because complex interdisciplinary societal problems are seldom fully defined the problem should be defined first before interventions can be considered. In several sessions of alternated individual preparation and group sessions the multi-disciplinary team will try to formulate the conceptual model and with this define the problem. In order to prevent group think on moments of discussion and brainstorming, groupware software is used to guarantee anonymous information exchange. The co-operative information exchange takes place in a Group Decision Room, and the whole process is supported by a facilitator. The method COMPRAM is developed by DeTombe ©.

Further research

Until now the method COMPRAM is defined and described till the definition of the problem, the first sub-cycle of problem handling. In the next years the method will be further defined till it includes all the phases of the problem handling process from awareness of the problem till evaluation of the implementation.

References:

DeTombe, D.J. (1991). Developing problem solving skills. *European Journal For High Ability*, vol. **2**, 18-27.

Forrester, J.W. (1987). Lessons from system dynamics modeling.*System Dynamics Review*, vol.3, nr.2, 136-149.

Newell, A. & Simon, H.A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Sol, H.G. (1988). Information Systems Development: A Problem Solving Approach. *Proceedings of the Symposium on System Analysis and Design*. Atlanta.

Appropriate technology for a developing society: CAL for Basic Concepts in Chemistry

M.M. DE V. STEYN and R.A. HASTY, *Department of Chemistry and*
P.M. ALEXANDER, *Department of Computer Science and Information Systems*
University of South Africa, P.O. Box 392,
Pretoria, 0001, South Africa

The University of South Africa (UNISA) is the oldest (1946) and one of the biggest (120 000 students) distance teaching institutions in the world. Most of UNISA's students are located in southern Africa. For economic and geographic reasons, all aspects of teaching are handled by academic staff situated on a single campus. The method of communication with students is mainly by means of the postal service.

We need to use new methods or approaches to teaching that will benefit not only the student, but will leave more time for the teaching staff to improve the standard of tutoring in their specific subject areas. We also need to overcome third world's distance-education problems. These problems include:

* a sparse and dispersed population,
* a poor telecommunications network,
* students with varied cultural, technological and educational backgrounds,
* restricted financial resources resulting in limited options regarding hardware, and
* the expected increase in the number of young students that will result in an even poorer ratio of students to teaching staff than exists at present.

These are in addition to the normal disadvantage of distance education in that students feel isolated due to lack of interaction with lecturers and fellow students.

We have met these challenges by producing CAL that:

* does not rely too heavily on infrastructures outside our control,
* can be distributed to our geographically dispersed student body on diskette by post,
* motivates students by requiring active participation, and
* provides intelligent and individualized responses.

The study package for chemistry has consisted of study guides, prescribed text books and tutorial letters. In addition, a practical component is conducted over a limited period of time at a central venue. Distance teaching of a practical subject such as chemistry has many short-comings which can be overcome using computer-based learning. Animation is used to advantage to explain complex concepts as it is dynamic and visual and therefore students can see what happens; interactive problem solving gives practice; and simulations augment wet-laboratory experiments. We have developed our courseware package, "Basic Concepts", for first-year analytical chemistry students. This will be used to supplement the usual study material. The courseware is designed to run on standard, DOS-based microcomputers with EGA screens, equipment which is reasonably affordable and available throughout South Africa. This approach meets most of our teaching requirements. Our courseware:

* provides structured tutorials and detailed examples,
* generates questions randomly which require students to practice basic skills,
* accepts and analyzes a wide variety of student inputs, and
* gives detailed model solutions and appropriate feedback depending upon student input.

The results of pilot testing for a group of 27 students on campus are encouraging. Factors which were identified as being improved are increased motivation, mastery of concepts and active participation by students.

The Biology Sleuth: The Development and Evaluation of an Interactive Learning Environment.

Rebecca Denning and Philip J. Smith
Cognitive Systems Engineering Laboratory
Department of Industrial and Systems Engineering
The Ohio State University 210 Baker Systems 1971 Neil Avenue Columbus, Ohio 43210 USA

The Biology Sleuth is an interactive learning environment developed to give high-school students exposure to health science issues and medical problem-solving. Learning is explicitly supported in the following areas:

1. Increasing students' understanding of the scientific method and diagnostic reasoning, and helping them to develop skill and confidence in applying these methods.

2. Reinforcing and extending students' knowledge of concepts pertinent to the biological and health sciences.

3. Supporting "incidental" learning of certain skills (computer literacy; social and communications skills; algebra; reading graphs and interpreting data) and knowledge (geography; history).

The Biology Sleuth software has two components. The first allows access to factual information about several diseases which are genetic in origin or may be caused by nutritional deficiencies or environmental effects. The second component is a series of problem-solving exercises in which students must determine which diseases could account for a patient's symptoms.

Initial formative evaluations of The Biology Sleuth have been conducted. These evaluations have allowed the developers the opportunity to observe two classrooms of high school students using the software. These evaluations were video taped and segments will be available as a part of this demonstration to show students working through the problem-solving exercises.

The video tapes show that The Biology Sleuth was successful at encouraging students to participate in the formation of hypotheses, to discuss the methods used to generate those hypotheses, and to select additional data to be gathered. The performance of the students while working with the system raises two questions:

1. Should group members be solving problems individually at some point in order to maximize their development of the problem-solving skills involved?

2. How does the design of the system influence such group problem-solving behavior?

These questions should be addressed in future studies.

Speech Interactive Strategies Learning with Hypermedia System Assistance

SVETLANA DIKAREVA

Department of Russian Language
Simferopol State University, Simferopol, Crimea, Ukraine 333036

The animated debates on the last East-West Conference in Kiev have proved that modern computer aided education is characterized by swift changes of information technologies and concept paradigms. In information technologies, key positions are now occupied by Multi, Hyper, and Telemedia systems being actively used in creating educational courseware.These technologies significally expand the opportunities for learning speech interaction with computer.

In Simferopol State University, we have a three years reseach project for investigating cognitive and environmental principles of verbal communication and elaborating computer language usage courses on this theoretical base. Some results of the reseach work were published in (Dikareva, S., & Ilovajskaya, H., 1992; Ronginsky, V., Dikareva,S., & Ilovajskaya, H., 1993).

This poster presents a module LANA (Linguistic ANAlysis) of the linguistic educational-information system, which support learning speech strategies by access to an Interactive Knowledge Base (IKB).

Theoretical background. Speech activity is a complicated social and cognitive process based on a different type of knowledge: language, encyclopaedic and interactive. Language usage is determined by complex interrelation of these types of knowledge, and depends on opinions of communicants.

A distinguished russian scholar L.P.Yakubinsky (1986), in his paper 'O dialoguicheckoi rechi'(first published in 1923), mentioned 2 manners of language usage: a conscious choice preceded by 'a struggle of senses', and speech automatism (using speech stereotypes, or patterns). A person falls into 'a struggle of senses' when he/she is confronted with a speech problem. We mean situations in which one senses conflict about an answer to the questions 'What needs to be spoken (or written)'. To analyze such situations we use an approach to decision making developed at the field of AI (Underwood, John H., 1989). According to this approach making decisions is seen as a heuristic search among alternatives in a problem space. The decisions which determine one's speech behavior over some time is a speech strategy.

Module LANA's architecture. LANA's learning material is a collection of texts, pictures, schemes,and graphics. The IKB consists of interactive rules; dialogues types; dialogue diagrams; speech acts vocabulary articles; illustrative examples. Each dialogue type is connected with dialogue rules and diagrams. To solve a speech problem means to answer to some questions with the aid of the IKB. (what is desired, under what conditions, by means of what tools - spoken, written, or electronic, etc.).

In organizing Multimedia material, we use an authoring Hypermedia system (Dikareva, S., & Ilovajskaya, H., 1992). The system functions on the base of IBM PC/AT or compatible PC.

References

Dikareva, S, & Ilovajskaya, H. (1992), "Text world" and Computer Assisted Language Learning. In *East-West Conference on Emerging Computer Technologies in Education. Abstracts.*, Moscow, Russia, ICSTI, 159-160.

Ronginsky, V., Dikareva,S., & Ilovajskaya, H.(1993), Cognitive Model of the Russian Orthography Computer Course. In *Computer Technologies in Education. Proceedings of ICCTE'93*, Kiev, Ukraine, 179-180.

John H.(1989), Artificial Intelligence and CALL. In *Modern Media In Forein Language Education: Theory And Implementation*, Ed.Smith Lincoinnved, Illinois., 197-225.

Yakubinsky, L.P.(1986), O dialoguicheskoi rechi. In *L.P. Yakubinsky. Yazyk i ego funkcionirovanie.*, M., Nauka, 17-58 (in Russian).

Teachers Guide: Classroom Instruction Utilizing CBI

D. J. Dirksen
Teacher Education Center
University of Northern Colorado, Greeley, Colorado, 80639, USA

In the arena of elementary and secondary education there have been many technological innovations that have been introduced and placed into hiding in audio-visual closets. Computers and computer based instruction is next in line, after many technological innovations, to fall into disuse because of lack of preparation and guidance for the teaching population. In order for diffusion of the innovation to occur, teachers need to receive the necessary support for them to successfully utilize computer technology in the classroom. The computer can be a powerful educational tool if incorporated into the curriculum properly. With the advent of microcomputers teachers gained a valuable teaching tool, a tool that is able to instruct, involve, and motivate almost any student; unfortunately, the computer is also able to bore, infuriate, and confuse almost any student. What the computer succeeds in doing is determined by the quality of software being used and the expertise of the teacher who is trying to incorporate microcomputers into the curriculum.

With the use of appropriate software and a certain amount of teacher expertise, the computer can be an asset to many educational programs. If used properly, the computer can aid students in the mastery of curriculum content and teach skills such as: math comprehension, reading comprehension, computation, vocabulary, grammar, punctuation, brainstorming, problem solving, scientific investigation, sequencing, editing techniques, and building mental schemas. These and many other skills can be easily and effectively taught with the aid of computers. In teaching these skills, the computer, when properly programmed, is able to use multiple learning modes: bright contrast and movement enhance the visual; kinesthetic learning is brought about by the use of the keyboard; and group interaction facilitates auditory learning (Morrison, 1983). Because a variety of learning modes can be used during instruction, retention of concepts that have been presented may be improved.

Unfortunately, many teachers are ill prepared to take advantage of the many teaching and learning opportunities the computer could afford them. According to the Turner Broadcasting System (1990), 97% of the schools have computers but only 50% of the teachers know how to use them. One teacher expressed the problem quite clearly, "There could be some giant mistakes if we think technology alone is going to change things. We need a staff that looks at how a child learns and asks the question 'What learning do you want the child to do?'" (Schulz, 1991). In order to utilize the computer effectively as a teaching tool, teachers need to know how to use the computer and to be familiar with the many teaching activities that it will enhance. For these ends to be accomplished, teacher inservice needs to be provided in these areas.

The use of computer technology needs to be addressed by and exemplified by our public schools. The question, "Why use technology?" was addressed by Finkel (1991) in his text *Technology Tools in the Information Age:*

1) To prepare students for the world they live in.
2) To prepare students to be a part of the technological work force.
3) To increase productivity of teachers and students.
4) To teach problem-solving.
5) To make it easier for teachers to provide a variety of learning and teaching strategies to meet the individual needs of learners.
6) To give teachers more time to spend with individual students.

References

Finkel, L. (1991). *Technology in the information age classroom.* Franklin, Beedle and Associates, Inc.

Morrison, V.B. (1983). Language experience reading with the microcomputer. *Reading Teacher*, 36, 448-449.

Schulz, E. (1991). Putting it all together. *Teacher Magazine*, 4, 44-49

Turner Broadcasting System (1990). Education and technology. *CNN Special Series.* (Broadcast June 18, 1990)

Dwyer, D.C.; Ringstaff, C.; & Sandholtz, J.H. (1991). Changes in teachers' beliefs and practices in technology-rich classrooms. *Educational Leadership*, 48(8), 45-52.

Developing a hypermedia package on reproduction in plants, protists and fungi.

C. DONALD, M. SANDERS AND A. CRITCHLEY
Department of Botany, University of the Witwatersrand, Private Bag 3, WITS, Johannesburg, South Africa, 2050

M. DONALD
P.O. Box 29, Lanseria, South Africa, 1748

A hypermedia remedial package has been developed for first year students studying botany at the University of the Witwatersrand, who find it difficult to understand and apply certain fundamental concepts about organisms' life cycles. This problem is especially prevalent amongst educationally disadvantaged students, and is found in other courses on plant reproduction at both secondary and tertiary levels.

An action research approach has been used, comprising three stages:
1. A combination of research methods was used to identify specific learning problems of the students.
2. A hypermedia package was developed to remedy the learning problems identified. Formative evaluation was carried out as sections of the package were developed, and consequent modifications were made.
3. A summative evaluation of the package will be conducted during formal implementation.

The outcome of the first stage of the study was the definition of instructional objectives for the package, in the light of the learning problems identified. Elements of constructivism, meaningful learning, schema theory, and conceptual change formed the theoretical basis for the design. The instructional objectives are to promote deep level conceptual learning with large numbers of students by ensuring that:
- the relevance of the practical component to the theoretical component of the course is made explicit,
- students practise transferring and applying fundamental concepts to different representative organisms, divisions and life cycles,
- alternative explanations are provided (to those given in class, as well as some in the application),
- students are offered a conceptual model of reproductive mechanisms in order to expand, re-structure or practise using new knowledge structures.

The following broad instructional design strategies were used during the second stage of the study, in order to meet the instructional objectives:
- Traditional computer-aided instruction modes (tutorial, quiz) and hypermedia features were combined to provide a flexible, yet structured learning environment.
- Varying levels of detail of explanation and questioning, as well as a variety of explanations for similar concepts were used to cater for the diverse learning needs of the heterogeneous student group.
- A conceptual model for learning about reproductive mechanisms, with visual "cues" to show similar patterns for different organisms, was used within individualised interaction sequences.
- Fundamental concepts that were incorrectly understood by students were incorporated into feedback cycles, which differed according to students' responses to specific tasks.

Students' lack of exposure to computer-based learning environments, and a predominance of learners with poorly developed metacognitive skills, necessitated that:
- "high risk" features (such as the user-interface and content structure) be identified, developed and tested with students and staff early in the development cycle,
- cognitive demands of the hypermedia user-interface be minimised to enhance ease of use,
- a strong diagnostic research component precede and supplement design and development.

The Design of a Computer-Based English Composition Prewriting Assistance Program for use in the Classroom

William J. Dorin, Ph.D.
Data Processing and Information Systems
Indiana University Northwest, Gary, IN, USA

Susan L. Sebok, J.D.
English and Foreign Language
South Suburban College, South Holland, IL, USA

A great deal of research in the eighties focuses on using the computer as a tool in English composition courses. Many researchers (Flood, et al., 1991) maintain that the word processor has helped students in English composition because students are able to compose and revise essays, and share text with other members of the class. Selfe (1985) finds students reluctant to use the computer for more than the final draft of an essay. Higgins, Flower, and Petraglia's (1992) research focuses on the planning process or prewriting. They believe the collaborative effort makes students better writers. Because of this research the authors have undertaken the design and programming of a computer assisted prewriting program called CAPP.

Model Based Development

To develop CAPP, the authors sought a basic model that describes the planning and organizing techniques of writing compositions. Two main sources, Axelrod and Cooper (1991) and Sorenson (1992) provide the main ingredients for the model. By combining these models, a typical CAPP session guides the student through finding a topic, gathering information, discovering what is already known, and writing a thesis sentence. Because of the structure of the program, students can analyze their audience or assemble their research either before or after choosing a topic.

The overall design and implementation follows a typical menu-driven model. The top-level menu has five main choices: FILE, PROJECT, RESEARCH, EVALUATION, and HELP (see figure 1). The entire program centers around the idea of the composition being a project.

FILE	PROJECT	RESEARCH	EVALUATION	HELP

Figure 1-- Top-level Menu

The testing of this software is qualitative in nature to determine how students will react to mechanics and aesthetics of the program. The primary concern for the authors is to learn how the students interact with the program and to learn if any kind of logic error or failure to consider user needs have been overlooked. The primary question that must be answered is, "Will instructors or students use the program?"

Reference

Axelrod, R., & Cooper, C. (1991). *The St. Martin's guide to writing*, (3rd ed). New York: St. Martin's Press.

Flood, J., Jensen, J., Lapp, D., Squire, D. (1991). Roles for computers in teaching the English language arts. *Handbook of research on teaching the English language arts*. (pp. 536-541). New York: MacMillan.

Higgins, L., Flower, L., & Petraglia, J. (1992). Planning text together: The role of critical reflection in student collaboration. *Written Communication, 9*(1), 48-84.

Selfe, C. (1985). The electronic pen: Computers and the composing process. In J. Collins, E. Sommers (Ed). *Writing on-Line: Using computers in the teaching of writing*. Upper Montclair: Boynton/Cook Publishers.

Sorenson, S. (1992). *Webster's new world student writing handbook*, (2nd ed). New York: Prentice Hall.

Curriculum Delivery via Remote Means

CHRIS FEIL
State Education Manager (Vic / Tas), Apple Computer Australia

This paper will centre on a program called Electronic Classroom. It will also be hopefully possible to see the program in action via a live demonstration across Australia where a colleague of mine, Richard Owens, from the Apple office in Queensland will use the Electronic Classroom program to show the audience how the program works. During the course of the demonstration, we will be showing aspects such as an ability to read compact disks remotely as well as the ability to transmit graphics and quicktime movies.

In the state of Victoria, there is burgeoning interest in distance education or remote delivery mechanisms. The reason for this lies in the fact that given that the state is small by Australian standards, there are still some very large distances within it.

The program has been written in the Pascal programming language by Robert Crago who has a company called Revelation Computing in the state of Queensland.

In principle, it behaves like a very powerful electronic whiteboard which supports graphics transmission as well as Quicktime. It can connect upto five remote locations simultaneously thereby making it possible for a teacher to use it in a distributed teaching situation. The teacher can enable the program to accept student input which all of the other participating students can view.

The Electronic classroom arose out of a need to teach subjects in a situation where there were great distances between teachers and students. One of the main uses of the program lies in the LOTE or Languages Other English Area. For example, the program has been used in the teaching of Japanese in a situation where there are small groups of students and one teacher in geographically isolated locations.

The state of Victoria is in the throes of condensing the number of schools especially in the senior high school area. The object is to provide a smaller number of larger secondary colleges where a more comprehensive curriculum can be offered. A consequence of this is an interim arrangement where some of these colleges might have multiple senior campuses. In much the same manner as above, the Electronic Classroom program has been employed to teach classes across the campuses which might never have been able to run due to a shortage of teaching expertise or an insufficient number of students.

It has also been used to enable classes to be conducted between secondary colleges and feeder primary schools, thus enabling valuable links to be developed.

By the time this paper is given, there will be a community-based project called the NE telecentre . This is situated at Wangarratta (200km NE of Melbourne) and the people in the centre will be using this program to run a series of community based education projects situated in towns over 130km away from the centre itself.

Over 650 copies of the program have been sold as at February, 1994. In Victoria, there are over 200 users of the program.

References

Elliot,N. The Telematics Manual Victoria (1991) ISBN 0 7241 9528 9
Hill,R.,Meulenberg, P., McNamara, S., & Dewildt, J., (1991) " ..It doesn't just happen!..." A Report of the Ministry Telematics Network Project

Distance Education
At California State University, Chico

Kathy A. Fernandes
Microcomputer Consultant
California State University Chico, Computing Services, Chico, CA 95929-0407

Distance education (DE) is a concept that has been around for sometime. Certainly the concept of a "book" originated so that information could be communicated to people throughout time and at a distance from the information's original source. DE spans both these dimensions: time and distance. Budget problems are forcing education to think about new paradigms for teaching more people with more diversity, but with less money and fewer resources. As technologies become more affordable and the capabilities ever-increasing, more attention is being brought to this field.

Flexibility, equality, cost efficiency and cost effectiveness are some of the advantages that DE has to offer. For students in the rural communities and students in big cities, avoiding the commute saves both time, money, and frustration. The flexibility of learning in the students' own environment makes distance education appealing to students. Students who need to stay closer to home to manage a family or to supervise employees can now have the opportunity to educate themselves. No longer are the students reliant exclusively on the education that is available within their own area. DE gives students the possibility to be taught in their field of study by some of the best teachers around the world. Universities no longer need to have a faculty member teach 4 sections of the same class. Less real estate is needed to teach a larger number of people and maintenance costs therefore decrease.

Simplistically there are seven tools used to achieve DE: print, radio, phone, TV/monitor, playback devices (e.g. tape recorder, VCR), computer, and FAX machines. Each of these devices can be used with different "delivery methods" to create a rich learning environment. Methods include the use of ISDN, tele- and videoconferencing, satellite, and multimedia. The use of these tools and methods are changing rapidly and allow unlimited possibilities for the future.

California State University, Chico is using five different methods of DE on campus. First, the ITFS Program, Instructional Television for Students, has been serving 16 sites in Northern California since 1975. Over 650 courses have been offered in 40 disciplines to the 11,159 students enrolled in the off-campus ITFS program. Second, Chico State also the SEN Program, Satellite Education Network, which delivers courses for a BS or MS in Computer Science via satellite to 14 corporate sites in 12 states. Many of the participants in this program are employees from companies such as Alcoa, Hewlett-Packard, IBM, MCI, and Pacific Bell.

Two separate software products are being used on campus to promote DE and develop interaction in the classroom. CUSeeMe is a free program developed by Cornell University that allows a Macintosh equipped with video capabilities to view a colleague over the Internet. FirstClass is a "groupware" product which is used by students in Physical Education and Recreation courses to send e-mail to one another, the instructors, and teaching assistants. FirstClass also allows conferencing on particular subjects. and uploading/downloading Hypercard and multimedia files for use in their projects.

The last DE method being developed on campus is curriculum for CSU's Project DELTA - Direct Electronic Learning and Teaching Alternative. Curriculum modules are being developed to run with multimedia, ISDN, and/or videoconferencing.

Chico State has an outstanding reputation for doing DE over the last 20 years. Still today Chico State continues to develop new ways to do DE with the new technology as it becomes more affordable and more available.

Hooked on Hypermedia

Susanne Flannelly
Clifton T. Barkalow School, 498 Stillwells Corner Rd., Freehold, N.J. 07728

Introduction:

Today, tomorrow, and for tomorrows after that, our students and ourselves will be bombarded with a continuous and relentless stream of massive amounts of information. To turn these quantities of material into usable knowledge, we have to be able to organize it in such a way that permits us to retrieve information efficiently. HyperStudio, HyperCard, and HyperScreen software programs are information organizing tools, that enables our students to present information effectively. Using hypermedia, students are able to relate text, scan images, draw, and include sounds for original and interesting presentations in a variety of subjects.

Background:

Eighth grade students in Freehold Township School District have successfully used these hypermedia programs over the past four years. To focus and motivate students interest, a student project is to design a stack that is based on one of the topics covered in the fourth grade science or social studies curriculum. The students plan, revise, and test their individual stacks until they are error-free. Then, they demonstrate their stack to a group of fourth grade students in our district. A copy of their stack is left with the fourth grade teacher so it can be used during class time for review. The eighth grade students print-out each screen and put it in booklet form to take home. Stacks are also videotaped so they can show their programs at home.

Educational Advantages:

In the process of planning and creating a stack, students develop and refine skills for problem solving, goal setting, analyzing data, synthesizing ideas, thinking in a non-linear way, organizing information, thinking visually, making decisions, developing a point of view, designing, testing, and revising a system as well as locating, gathering, and evaluation information.

Using hypermedia programs has enabled students to realize the importance of computers as a tool. They find that using the computer is a highly motivating and develop a greater appreciation for the skills involved in creating stack presentations. Student self-esteem rose as they realized near professional results achieved in their final projects. They were able to analyze their perceptions about their work and realized that one of the primary goals of computer education is to effectively integrate information assimilated in a computer class across the curriculum.

Conclusion:

This project has proven to be effective. It has enabled students to understand the importance of the computer as a tool. This was demonstrated by students realization of the value of the experience by applying what they learned to other areas of study. They are hooked on hypermedia.

Acknowledgements:

My thanks to my eighth grade computer students whose enthusiasm make it a pleasure to teach. Also, thanks to William Canning and Robert MacMillan for their support.

Inside the T9000: Hypermedia Explanations of a Parallel Microprocessor

G. FULANTELLI, M. ARRIGO
Istituto di Tecnologie Didattiche e Formative
Italian National Research Council (CNR) - Via R. Settimo, 55 - 90139 Palermo - Italy

Continuous advances in VLSI technology have recently made it possible to produce more and more complex microprocessors and, as a consequence, the need has arisen for innovative documentation tools. Conventional purely textual technical manuals, even if supported by several graphics, fail to clearly explain the dynamic behaviour of these devices.

The key concept of the illustrated work is to combine a hypermedia information framework with interactive graphic animations to simplify the educational and training activities on microelectronic devices. The Hypermedia approach has been adopted to develop an original form of technical documentation for the INMOS T9000 transputer (INMOS is a member of the SGS-THOMSON Microelectronics Group). The T9000 is the last member of a family of parallel microprocessors which are normally used to build distributed computers. To complicate documentation issues further, it can simultaneously perform internal parallel operations and exchange messages and data with similar devices; this makes its description very difficult.

The developed system consists of a fully interactive graphic model of the T9000 which can be explored to observe the architecture and the working mechanisms of the transputer. Dynamic behaviour is made much clearer by interactive graphic animations integrated in the model. The modelling of the T9000 has been performed through multiple levels of abstraction of the real physical architecture of the system. At the highest level, we consider the T9000 as a whole; at the next level, the T9000 is structured as a set of subsystems, which in turn can be broken down to obtain deeper and deeper detail levels. Consequently, at increasing detail levels there are correspondingly more and more accurate explanations and animations.

User interaction is based upon a direct manipulation of the objects shown on the screen; these can be labelled boxes representing the model circuital components or buttons providing users with tools to control navigation. Components are explored by clicking on the corresponding labelled boxes. A backtracking button provides a link to the previous abstraction level.

An animated guide acts as the presenter of the microprocessor. Users can watch the guide at work on the screen, while it explains - through typical balloons - the component or animation of their choice. Consequently, the developed hypermedia application resembles a fully interactive comic-strip story of a rather sophisticated subject such as the INMOS T9000 transputer documentation.

Vocal comments can be activated through a "microphone" button during lessons and animations, in order to have more details on a component or mechanism.

The user interface has been designed in such a way that users are not aware of the hypermedia structure of the documentation. Yet, the fully interactive model adopted by the system permits users to establish their own navigational paths through the model: at each level, users are allowed to explore components in the order they prefer. As a consequence, users have complete control over the presentation order. A pleasant side effect of the multilevel model is that the system can be used by users with different knowledge requirements. In fact, users can stop their navigation at the wished detail depth.

In the opinion of the authors, hypermedia presentations, such as the illustrated system, can be used as effective communication media to document microelectronic devices and, consequently, to reduce technical training costs and teaching efforts.

Acknowledgements

The authors wish to thank Prof. Giovanni Degli Antoni for his invaluable guidance and Luciana Natale for her crucial support. They would also like to express their sincere gratitude to the INMOS T9000's team of architects, to SGS-Thomson and to CoRiMMe, for their support and co-operation.

Using the A* Search Space to Develop a General-Purpose Intelligent Tutoring Shell

THOMAS P. GALVIN and NEIL C. ROWE
Department of Computer Science
Naval Postgraduate School, Monterey, California 93940

Many applications for an intelligent tutoring system involve problems that require quantitative judgment -- for example, scheduling, automotive repair, and manufacturing design. We wanted to use a single paradigm to model these problems so we could build a library of lesson material on different subjects while minimizing resource requirements. The solution we found is a tutoring shell, called ASTutor, using the A* search algorithm. A* has the advantage of being able to handle the widest range of search problems. Problems that can be adequately solved using best-first or branch-and-bound can be solved in A* by eliminating the use of some A* functions. Conversely, we would be unable to model an A* problem using best-first or other search techniques.

Overview of the ASTutor System

ASTutor is a problem-independent interface shell written in Quintus Prolog and provides the main command loop, standard input and output routines, and general-purpose tutoring rules to detect a student's errors. ASTutor problems can be compiled into an executable file, or interpreted from within the Quintus Prolog system.

Teachers build lessons by writing Prolog rules for problem-dependent A* search routines (goal description, cost and evaluation functions), database management requirements, and input/output routines. The database management requirements are the lesson-specific facts and rules. For input/output, the teacher must prepare templates for displaying the current state and providing means for a student to query for more details about some aspect of a state or potential actions to take. We do a qualitative comparison of actions ("better", "much better", etc.) since numbers produced in an A* search will rarely mean anything to the student.

Upon running an ASTutor problem, the student will receive an introduction to the problem along with a display of the initial state. He then must perform an A* search towards the goal, aided by ASTutor. At each step, the student must choose from a list of possible actions. If the action does not appear to be the best action based on the results of the problem-dependent A* routines, ASTutor will notify the student. The student may try a different action, continue ahead, place a bookmark on the current state while he explores other possible actions, backtrack, or prune known incorrect actions. He continues to run the problem until he reaches a goal state.

Case Study of Decryption and Future Research Directions

Our primary case study is the problem of decoding a monoalphabetically encrypted string. This is a good case study because of the large number of choices a student has at each step. It identified some key problem areas to explore: First, that we must provide a system of tolerances so that ASTutor will accept approaches not siginificantly more costly than ASTutor's solution. Second, that we must build checking and heuristic-naming routines to ensure the cost and evaluations functions are adequate. Third, we need to build lesson-construction tools in order to hide the Prolog code implementation from the teacher.

We are exploring combining ASTutor with means-ends analysis based tutoring. This will afford us the ability to quickly evaluate both the student's choice of action and the student's choice of parameters. We have built a preliminary lesson-building program for teachers which uses object-oriented modeling of the scenario and provides checking routines to ensure a lesson description is both complete and consistent.

Neuroinformatics at the Secondary and Post-Secondary Levels of Education in Lithuania

Algis GARLIAUSKAS
Neuroinformatics Department,
Institute of Mathematics and Informatics,
Akademijos 4, 2600, Vilnius, Lithuania
Ph.:(370 2) 359608, Fax.:(370 2) 359209
E-mail: Algis.Garliauskas@mii.lt

The traditional informatics is based on the paradigm of sequentional computations to evaluate functions and solvings. Today a wider and more coplex notion is necessary. The computer is to be seen as a component in a distributed, interactive systems of human, computers, machines and other artificial and natural dynamical systems such that the paradigm of a group of communicating individuals is more adequate. In particular the areas of artificial intelligence and of artificial neural networks are considered to be subareas of informatics. Therefore the neural computational linquistics and cognitive science are very important for engineering, communication theory and cybernetics as well as for education processes at the secondary and post-secondary levels. The main aim is to get to know pupils and students new ideas of informatics: neural models, neural networks, parallelism in computing, learning with/out superviser and recalling, introduction in neuroinformatics, neural network software applying in different practical areas.

The main objectives of this activities are following:

a) to introduce a new area of informatics, i.e., neuroinformatics for older classes of Lithuanian secondary schools and students as well as in other Baltic Countries;

b) to create simple with high visualization education neural network software and to promote in practical classes of the secondary and post-secondary levels of education;

c)to use wide the computer-electronic networks for education process together with new ideas of neuroinformatics;

On the basis of the teaching of informatics (Bauer, Goos, 1982, 1984; Brauer, 1987) and the neural network software made by NeuralWare, Inc. (Klimasauskas, 1989) and ourselves we have been created the neural network introduction curriculum for older classes of the secondary schools. This course will be promoted by computer network using the electronic mail principles.

References

Bauer, F.L., Goos, G. (1982, 1984). Informatik, eine enführende Überbsicht, 2 parts. 3rd edition, Springer - Verlag, Berlin.

Brauer, W. (1987). Informatics and Computer Science in Education, in: Informatics and the Teaching of Mathematics. Elseviers Science Publ.B.V.,Amsterdam, 1987.

Klimasauskas, C. (1989). Neural Computing. Vol.1, Copyright, NeuralWare, Inc.

Computerized Testing Using an Authoring System

M. L. GLOWACKI, AND A. C. MCFADDEN
Laboratory for Interactive Technology
The University of Alabama, Tuscaloosa, Alabama, 35487, USA

Currently available computer software authoring packages offer many options for the development of educational applications. One such application is that of computerized testing. Considerable research has been conducted regarding the feasibility, advantages, and effects of using computerized testing in educational settings (Bugbee & Bernt, 1990; Legg & Buhr, 1992; Mizokawa, 1991; Powell, 1992). Although much research has been conducted regarding computerized testing, its use in the classroom has not been widespread. This may be due to the fact that many educators do not have the programming skills necessary to develop this type of application (Mizokawa, 1991). An alternative for those who lack the necessary programming skills and/or the time to learn a programming language is one of the many authoring packages currently available. One authoring package, *Authorware Professional*, employs an object-oriented interface and has been used in the development of a computerized testing system.

The purpose of this presentation is to demonstrate the use of multimedia technology using the authoring package, *Authorware Professional*. The computerized testing system that has been developed using *Authorware Professional* enables the user to develop many different types of testing formats and generate equivalent tests with randomly selected items. The tests are computer-administered and utilize several types of formats including multiple choice, true-false, and short answer. Some test items include graphics, digitized video, and/or sound, and responses are given in several different ways, including pushbuttons, text, and use of the mouse. Testing systems have been developed for use on both the Macintosh and Windows platforms.

Advantages of using a computerized testing system have included more time for direct instruction and teaching, rather than using class time for exams; students are able to take the tests when they feel they are ready; tests can be adapted for students with disabilities (i.e., key pressing vs. mouse); instructors spend less time developing and correcting tests and more time in preparation and research for class instruction; students receive immediate feedback on test scores and incorrect items; the test can be stopped for students whose responses indicate they do not know the material, thus reducing frustration; different versions can be administered to discourage cheating; because items are given one at a time, it's difficult to use clues from other items; and test items can be added, revised, or deleted with little difficulty.

The advantages of using an authoring package to develop a computerized testing system include less learning time than that needed for learning a programming language; use of graphics, digitized video, sound, etc. within a test; ease of use; and because models can be developed and used over and over in various applications, less development time. These advantages make the use of an authoring package for development of computerized testing as well as other educational applications a feasible alternative for educators.

References

Bugbee, A. C., Jr., & Bernt, F. M. (1990). Testing by computer: Findings in six years of use 1982-1988. *Journal of Research on Computing in Education*, 23(1), 87-100.

Legg, S. M., & Buhr, D. C. (1992, Summer). Computerized adaptive testing with different groups. *Educational Measurement: Issues and Practice*, 23-27.

Mizokawa, D. T. (1991). Computer-managed testing in schools. *Educational Technology, 31*(1), 21-25.

Powell, Z. E. (1992, April) *Test anxiety and test performance under computerized adaptive testing methods.* Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA.

Distance Learners' Perceptions of a Computer-Assisted Learning Programme Designed to Enhance Their Decision-Making Skills Integral to Nursing Practice

D. GOLDSWORTHY
Faculty of Health Science
University of Central Queensland, Rockhampton, Queensland, Australia, 4702

Developing Decision-Making Skills by Using CAL

Decision-making experience in nursing practice is essential for nurses to develop skills in management. Yet, practitioners cannot normally acquire these skills except by experience in a management environment itself. A specially designed computer simulation programme in decision-making allows the development of these skills in such a situation without risk to student or client (Curry, Elliot, Wheeler, and Robert, 1991). In pursuing this point Lauri (1992) describes how computer simulations can be used as a means of assessing the decision-making processes of nurses who are working in public health. However, students need to feel comfortable with CAL in order to obtain maximum benefit from its teaching programme. Additionally the structure, content and presentation of CAL needs to complement the individual student's learning style. Most nurses are adult learners and, according to Knowles (1986) are orientated towards and motivated by learning which is relevant to their purpose. This is of especial importance to distance education nursing students, many of whom select distance education courses as a means of development.

Decision-Making Skills, CAL and Distance Education Nursing Students

Distance education programmes for nurses in Australia still depend mainly on printed text-based materials. Little has been done to make computer-assisted learning (CAL) available to these students, lack of research into learning styles and CAL compounding the problem. A study was conducted to investigate the students' perceptions of a CAL programme designed to enhance and develop their decision-making processes. The programme was based on theories of problem-solving and decision-making, and on knowledge of nursing management and leadership styles. The programme uses a case study simulation approach, presenting nursing management situations coomon to the health care environment. The distance education students responded positively to the simulations and the programme worked well. Two of the main conclusions from the study showed that the programme was effective as a learning strategy for the majority of the participants, regardless of their individual learning styles. The interactivity between the student and the computer programme combined with the feedback to the student, emerged as a factor crucial to the success of the programme. The main findings suggest computer simulation case studies can stimulate, enhance and motivate learning and be used as a means of promoting the development of decision-making skills for distance education nursing students. Other findings include:

- F1: CAL content, structure, format and presentation are more important than sophisticated technical or visual capabilities.
- F2: Written guidelines and lecturer support are a priority for distance education nursing students who are using CAL.

References

Curry, P., Elliot, D., Wheeler, E.; and Ghude, R. (1991). Implementing an Automated Care Planning System in a Nursing Curriculum'. *Computers in Nursing*. 9 (4), 139-142.
Knowles, M. S. (1984).*The Adult Learner: A neglected species*. 3rd edition. Houston: Gulf Publishing.
Lauri, S. (1992).'Using a Computer Simulation Program to Assess the Decision-Making Process in Child Health'. *Computers in Nursing*, 10 (4,) pages 171-177.

Using Distance Learning Technologies to Pro;ide Mathematics Inservice for Middle Grade Teachers

Terry Goodman
Department of Mathematics and Computer Science
Central Missouri State University
Warrensburg, MO 64093

The concept of distance education using information technology has been explored and found to be a viable substitute for traditional modes of delivering instruction. Distance education in such forms as correspondence courses and instructional television has been used for some time. Distance education via satellite has emerged only within the last decade with extensive efforts in the state of Alaska as early as 1983. Other agencies, such as Oklahoma State University have taken the lead in the development of this form of distance education.

Information technology, as it relates to distance education, continues to develop with dramatic speed. One of its major contributions is to provide training and staff development in locations where formal instruction, expert teachers, and sorely needed resources are difficult to obtain. Distance and lack of resources make it difficult for many teachers in Missouri to receive continuing education. The major goal of this project was to make use of information technology to update and revitalize the training of middle grade mathematics teachers.

Overview of the Project

In this project, a one semester hour, graduate course was developed using a video tape format and broadcast via satellite to 19 school sites in the state of Missouri. The focus of the course was on identifying strategies, activities, and resources useful in teaching middle grade mathematics. The course consisted of eight meetings, one per week, of which six were broadcast sessions. The mathematics content of the course focused on estimation, statistics, probability, geometry, patterns, and problem solving.

For one-half hour before the start of a broadcast, participants engaged in a variety of activities at individual Host Sites. At each site there was a designated Site Facilitator who directed discussion of activities that the teachers had used with their students the preceding week. The Site Facilitator also provided introduction to the topic(s) to be investigated during the broadcast session to follow, provided the participants with materials they would need, and answered questions concerning the topic(s) of the previous week's broadcast.

For one-half hour after the one hour broadcast, the participants worked on homework and other assignments and discussed the broadcast session they had viewed that day. The Site Facilitator clarified homework and other assignments, monitored each teacher's portfolio, and served as a resource person during this time.

The broadcasts were interactive in that a school site could call the broadcast facility on a designated toll-free line and ask questions about the current lesson or seek other information regarding the course. These questions were answered "live" at specified times during a broadcast.

The six video tapes were planned, scripted, and narrated by eight TV Presenters, each of whom was a practicing middle grade mathemtics teacher. Resources, technical expertise, and production facilities were provided by Central Missouri State University while broadcast faciltities were provided by The Missouri State School Boards Association. Each video taped lesson contained the following components.

1. Two or three TV Presenters introducing and developing the topic for that particular lesson
2. Computer generated graphics used to illustrate and amplify the mathematics concepts for that lesson
3. Vignettes showing middle grade teachers and students in classrooms engaging in a variety of activities
4. Vignettes showing how the mathematics from the lesson is applied in "real-world" settings

Acknowledgements

This project was funded by the Dwight D. Eisenhower Mathematics and Science Education Act

Distance Education: Interactive Video Conferencing in Music Instruction

PETER GOUZOUASIS
Department of Visual and Performing Arts in Education
University of British Columbia, 2125 Main Mall, Vancouver, BC, V6T 1Z4, Canada
E-Mail: petergou@unixg.ubc.ca

DAVID WALKER
Commonwealth of Learning, 1700-777 Dunsmuir Street, Vancouver, BC, V7Y 1K4, Canada
E-Mail: dwalker@unixg.ubc.ca

LEONARD BURTENSHAW
University of Sydney
Sydney Conservatorium of Music, Macquarie Street, Sydney, NSW 2000, Australia
E-Mail: leonardb@extro.ucc.su.oz.au

Communications media—in the form of radio, television, motion pictures, telephones, and computer controlled information systems—have defined modern culture on a global level since the 1930s. Burtenshaw's (1993a, 1993b) ideas about distance education and previous experiments with video conferencing have provided us with a glimpse of how video conferencing technology could revolutionize music instruction transcontinentally. Rather than fully-interactive sessions, where participants are able to immediately respond in an echoic (rote) or sequential manner to a teacher's instructions, most video conferencing experiments involve the observation of teacher-student interactions in live teaching situations. In a distance education context, Gouzouasis (1994) has successfully experimented in a limited interactive context with preschool children. Based on that study, he concluded that even though video resolution quality was not optimal with a two line transmission, video conferencing is a cost effective communications technique and distance education tool.

With the intent of exploring mass media in education, the purpose of this live demonstration experiment will be to learn more about the use of video conferencing in a music learning environment. Specifically, the the primary problem will be to examine the efficacy of video conferencing—between sites in Vancouver, British Columbia and Sydney, Australia—in a multifaceted, interactive, music instruction segment with preschool children. Concomitant with the purpose, the secondary problem will be to further develop objective techniques for the evaluation of teaching techniques and materials in video conferencing instructional sessions.

This work is important from a number of perspectives. First, although music is an essential component of multimedia, visuals are usually the emphasized media of expression. Second, children's educational programming does not adequately provide a broad base of music information to young children. Moreover, the developmental appropriateness of televised materials may be questioned on a number of levels, especially in a music education context. Third, video conferencing applications in distance education tend to lack total interactivity between presenters and participants. It is from that perspective that the researchers wish to explore the full potential of video conferencing technology in an interective setting that requires the transmission of sophisticated acoustic and visual information.

References

Burtenshaw, L. (1993a, April). *Perspectives of distance learning in Australia: The potential role of video conferencing in music education.* Invited paper at the Suncoast Music Education Forum, University of South Florida, Tampa, FL.

Burtenshaw, L. (1993b, January). Video conferencing--its implications for music education. Invited paper at the Colorado Music Educators' Conference, Colorado Springs, CO.

Gouzouasis, P. (1994, in press). *Video conferencing with preschool children: Mass communications media in music instruction.* Refereed paper for ED-MEDIA 94--World Conference on Educational Multimedia and Hypermedia. Vancouver, British Columbia.

7 1 U

PRABODH: An Intelligent Tutor for Teaching Language Skills to Young Children

REKHA GOVIL AND MADHAVI SAXENA
Department of Computer Science
Banasthali Vidyapith -304 022, Rajasthan, INDIA

Intelligent Tutoring System (ITS) is characterized by inclusion of three kinds of expertize, domain knowledge, knowledge of students learning behaviour and tutoring strategic knowledge. The programmes incorporating these expertize are embedded in a sophisticated instructional environment which facilitates tutorial communication.

PRABODH is an ITS designed for teaching Hindi language skills to young children to provide them meaningful practice to develop the roquisit level of automaticity in grammar skills. It is an authoring system which allows a teacher to construct individualized tutorials, drills & tools for students.It can be easily integrated into the curriculum of primary level language teaching and learning by practicing teachers.

Description of the framework

It has six components. *The domain knowledge component* contains the knowledge of Hindi grammar appropriate for primary school teaching. The sequencing of this knowledge is maintained by a topic network where a topic represents a unit of knowledge that can be taught.

The strategic knowledge base, acquired from teacher experts, contains knowledge of how to teach. PRABODH has a structured curriculum. There are lessons linearly structured but topics in them are componentially structured. The tutor module facilitates acquisition of strategic knowledge from the tutor and presentation of the information to the student and acquiring students' responses, by - *tutor* interface, student *interface and tutor engine* respectively.

The student model component keeps track of students' knowledge and needs through diagnosis and modelling.It allows to detect the students' misconception or missing conception and then through the presentation of right remedial material the errors are overcome. The student model also has a *learning component* to incorporate learning new facts about students' behaviour from their responses.

Beside teaching Hindi as mothertongue it can also be used to teach Hindi as a second language or foreign language by the inclusion of right kind of tutorial material.

References

Polson, M.C. & Richardson J.J. (Eds) (1988). *Foundation of Intelligent Tutoring Systems.* Hillsdale N.J. : Lawrence Erlbaum Associates.

McCalla,G.I. & Greer,J.E.(1990). *Understanding literacy & cognition: Theory, research & application.* New York: Plenum Publishing.

Developing Effective Simulations for Problem Solving

M. GREDLER, K. BARANICH
College of Education and Southeast Manufacturing Technology Center
University of South Carolina, Columbia, South Carolina 29208 USA

Simulations provide an environment in which learners interact with a knowledge domain in ways different from that of either direct or tutorial instruction. However, many computer programs which purport to be instructional simulations are instead a type of adventure game or simply graphic representations of events or processes. Essential criteria for simulations are identified by Gredler (1992). First, simulations for instruction are dynamic evolving exercises in which the student accepts a functional role and/or responsibilities and interacts with other individuals or with a complex evolving situation. Second, simulations involve the ongoing interactions of at least two variables according to a verifiable qualitative or quantitative model that relates the changes in one variable with changes in others.

Types of Simulations

Analysis of the relationships between the learner's role and other variables indicates two major types of simulations (Gredler, in press). They are (1) experiential and (2) symbolic. Experiential simulations establish a particular psychological reality and place participants in defined roles within that reality to interact with an evolving scenario. Examples of experiential simulations include diagnosing and managing the treatment of a comatose patient in the emergency room, managing the finances of a bank or corporation for several business quarters and interacting with one's neighbors concerning the proposal of a nuclear power plant to locate in the village. In contrast, symbolic simulations are intact data bases, complex systems (e.g., an ecological system or complex equipment system), or sets of processes in which students conduct research on the data base or attempt to discover key components of the system or set of processes. In an experiential simulation, the student is a major component of the situation whereas the student operates on but remains external to the symbolic simulation. Experiential simulations, in other words, provide students with opportunities to practice complex professional roles whereas symbolic simulations provide opportunities for developing mental models and conducting complex projects.

The creation of effective simulations is a complex task, much more difficult than preparing a classroom lecture. However, their use adds immeasurably to the development of the students' repertoire of higher order thinking skills including those such as cognitive strategies, decision making, and problem solving.

References

Gredler, M. (1992). *Designing and evaluating games and simulations: A process approach*. Kogan Page. London.

Gredler, M. (in press). *Games and simulations: a technology in search of a (research) paradigm.* In D. Jonassen et al. (eds.) Handbook of Research on Media and Technology.

742

Infusion and Transfusion of Instructional Technology at NAU: Exploring Models for Professional Development in Higher Education

C. GUNN
Center for Excellence in Education
Box 5774, Northern Arizona University
Flagstaff, AZ 86011-5774

The Transfusion for Infusion Model

The Center for Excellence in Education (CEE) at Northern Arizona University (NAU) prepares teachers, school administrators, educational psychologists and counselors for our nation's schools. CEE's Mission Statement includes a technology initiative which calls for the infusion of technology by faculty into all discipline areas, where educational technology is modeled for undergraduate and graduate students in all courses where appropriate. With most faculty teaching full time and with overloads, with growing student numbers causing increased class loads, and with research and service obligations, most faculty have not had the time required to learn new technologies or develop classroom applications supported by technology.

The CEE Technology Committee received a 1-year grant which funded release time (transfusion) for one faculty member from each of the following areas: special education, instructional leadership (elementary & secondary), educational leadership, educational psychology, bilingual and multicultural education, and instructional technology. Each of these seven faculty members were granted 1/4 release time for two semesters for staff development in the area of multimedia. College administrators have taken seriously the recommendations of their faculty to provide release time in technology development and provided this release time as their match to a university-funded technology grant. The grant participants believed it was through the transfusion of time that infusion of technology would take place.

Using a trainer-of-trainer model, the faculty spent the first grant semester learning a particular application or portion of multimedia (i.e., still video camera, interactive video, HyperCard) and are teaching other grant participants that application. Participants are applying that knowledge by developing an application for their particular area or discipline, and will demonstrate the direct application to classroom instruction to colleagues in that area. This development process is documented throughout the project by journals and through a tracking model for innovations in technology. Four seminars are planned during the second semester: three instructional seminars for CEE faculty on instructional design and goal setting. A fourth general session on multimedia equipment available at CEE has been held for all interested NAU faculty. The grant participants will also write working papers which will be published and distributed to CEE faculty, then to any interested NAU faculty.

Two areas for future study have emerged. 1.) Participants report that release time for technology learning and development is not a particularly ideal solution when administrative priorities do not also include finding teaching replacements. Six of seven participants are participating in the project as an overload and time still appears to be a critical issue. All 7 participants have continued their involvement to various degrees, and overload compensation has been appreciated. Real release time, however, will be sought for future projects. 2.) Goals set by individuals with guidance from educational technology faculty, rather than the goals outlined in the project, also became important for successful participation. Just as with our preservice teachers, amount of familiarity and comfort with technology available determined the extent of participation by faculty on the grant. Three faculty participants, who were minimally involved with instructional technology before the grant, worked with an educational technology faculty member to find an appropriate place for themselves in this project--finding software applications and infusion strategies that worked in their content areas. All other participants spent their time in learning to use multimedia hardware and software. On reflection, it may be these three faculty who were able to move the farthest in technology infusion and to impact student use of technology integrated into course projects the most.

Distance Education Technology:
A Course for Teachers and Librarians

Curtis P. Ho, Ph.D
Department of Educational Technology
University of Hawaii at Manoa
Honolulu, Hawaii 96822

Description

Distance Education Technology is a course offered by the Educational Technology Department at the University of Hawaii at Manoa. The course was developed to serve two populations: teachers and librarians; therefore the course was cross-listed with the Graduate School of Library and Information Studies. The goal of the course was to explore various technologies used for delivering and retrieving information for school-based and distance instruction. Voice, data, and video served as the topics as well as the mechanisms for interactive delivery of course content and student projects. Emphasis was placed on the application of technologies in Hawaii's public schools. The course was delivered live and interactively over the Hawaii Interactive Television System (HITS), a 2-way audio/2-way video system. Class activities included lectures, guest speakers, demonstrations, mediated instruction, class discussions and team presentations. Assignments were telecommunicated to the instructor using fax and electronic mail. Class projects were presented by students live over HITS.

The coursed aired daily for 3 weeks during the 1993 summer. There were 45 students enrolled in the origination site and 22 enrolled at 5 receive sites on 4 neighboring islands. A combination of fiber-optics, coaxial cable, microwave and ITFS technologies were used to broadcast the course statewide over HITS. All students were given Internet accounts and used modems, videotex terminals, computing center terminals, and computers on ethernet LANs.

Course Objectives

1) define terminology and acronyms of technology used in education.
2) describe major characteristics of information technologies and analyze their strengths and weaknesses in delivering school-based and distance instruction.
(3) describe applications of technology being used to solve instructional problems in Hawaii's schools.
(4) discuss implications of telecommunication systems and networks in providing and accessing information from libraries, government and other public resources.
(5) discuss potential problems educators face in designing, producing and delivering direct instruction and resource information over telecommunication systems.
(6) write a grant proposal for improving the accessibility or delivery of direct instruction, resource information, or communication at a school or resource center through technology.
(7) present grant proposal orally and visually over HITS.

Outcome

A course evaluation confirmed the appropriateness of the content to both teachers' and librarians' needs. There were no differences in overall course ratings between on-campus and receive-site students. The use of electronic mail was clearly identified as a critical component for the course's success. The grant proposal was another component of the course receiving especially positive ratings. Students worked hard on their proposals. A number of proposals were submitted for actual grants and one of them was awarded $44,000 by a school district in Honolulu.

The Effects of Videodisc Training on Student Learning

DIANE B. HOADLEY, J.D.
Associate Professor of Business Law
University of South Dakota, School of Business, 213 Patterson Hall, Vermillion, SD, USA 57069

MICHAEL R. HOADLEY, PH.D.
Director of the Center for Interactive Technology in Education and Corporations (InTEC)
University of South Dakota, School of Education, 414 East Clark Street, Vermillion, SD, USA 57069

Purpose of the Study

The purpose of this study was to determine the effect of using videodisc-assisted instruction on student learning in a business education classroom as part of the regular curriculum offering.

Methodology

Subjects were drawn from the four sections of BADM 350: "Legal Environment of Business" offered during the 1993-94 school year at the University of South Dakota. There were 69 students in the experimental group (two sections) and 46 students in the control group (two sections).

A 20-item knowledge test was developed by the investigators to reflect the objectives for the section in the course which deals with "Contracts". The instrument was tested to determine its readability and clarity and was reviewed by School of Business and Law School faculty at the University of South Dakota for content validity.

The study took four class periods to complete the unit, which spanned a two-week period. The control group received coverage of the material using traditional lecture methods. In addition to the lecture method, the experimental group received an intervention consisting of commercially prepared videodisc supplements presented by the instructor.

Results of the Study

Statistically no differences ($p \leq .05$) existed between the two groups at the time of pretest, so the groups were considered comparable. Comparisons were then made between the groups to determine the effect of the experimental intervention. Statistical analysis of the group data was performed using paired t-test and analysis of variance. Results revealed no significant differences between the control and experimental groups ($p \leq .05$).

Findings and Recommendations

Results indicate that the use of technology may not always improve student learning, no matter how well-developed the medium. Two factors which must be considered and which warrant further investigation are the ability of the instructor to use the technology effectively (which involves training, familiarity with the equipment, and ease of use) and the role of students in learning (active vs. passive) when technology is used.

Findings of this study could prove to be significant in further development of the business law curriculum at the University of South Dakota and other institutions which prepare business professionals. Decisions must be made relative to the emerging role of technology, but the pedagogical value of the technology must also be determined. Ultimately, innovations in the use of technology must lead to better student performance in terms of their comprehension, as well as their application of the material to real-life situations.

Evaluation of a Video as Distance-learning Material for Dental Practitioners

R D HOLT (1), D C RULE (2), R M BASKER (3), J C DAVENPORT (4), J P RALPH (3), J J MURRAY (5) AND K A EATON (6)

(1) Institute of Dental Surgery, 256 Gray's Inn Rd, London, WC1X 8JD, UK.

(2) BPMF, 33 Millman St, London, WC1N 3EJ, UK.

(3) Leeds Dental Institute, Clarendon Way, Leeds, LS2 9LU, UK

(4) The Dental School, St Chad's Queensway, Birmingham, B4 6NN, UK. (Principle presenter).

(5) Dental School, University of Newcastle, Framlington Place, Newcastle upon Tyne, NE2 4BW, UK.

(6) Department of Health, Room 326, Richmond House, 79 Whitehall, London, SW1A 2NS, UK.

Conventional dental postgraduate courses often require practitioners to give up clinical sessions and to travel a significant distance to a postgraduate centre in order to attend . These barriers are common reasons given by practitioners for not attending such courses. Distance-learning methods overcome these disadvantages. Therefore in 1986 the UK Dept of Health commissioned the British Postgraduate Medical Federation (BPMF) to produce educational videos for dental practitioners to study in their own homes. The videos were planned by a steering committee of general dental practitioners, dental academics and postgraduate advisers. Dental specialists produced the scripts but the final results were very much influenced by the general dental practitioner (consumer) advisers. Professional television teams were hired to produce the videos which were then distributed free to all principle dental practitioners working in the UK General Dental Services. Many of the videos were accompanied by an illustrated booklet which expanded on the topics covered in the video and also provided a bibliography to guide further study. It has been reported that a large proportion of those receiving the videos perceived them to be of value[1]. However, a formal evaluation of individual videos had not been carried out prior to the present study.

This study was set up as a joint initiative between the BPMF and the Committee on Continuing Education and Training of Dentists (COCET) and was funded by the Dept. of Health. The video ('THE ALLIANCE") which formed the basis of the evaluation was concerned with the design of dentures and was produced by three of the authors (RMB, JCD & JPR) in 1991. The investigation was carried out through two postal questionnaires sent to a random sample of 636 dentists in England and Wales before and after watching the video The questionnaires covered the dentist's profile, method of designing dentures, and preferred options from a variety of denture designs. The objective was to identify any change in design practice resulting from watching the video and also to record the dentist's opinion of the usefulness of the video.

Both the pre- and post-video questionnaires were completed and returned by 257 (40%) of the practitioners. The vast majority of those (82%) felt that the video had been useful. Intentions to improve design practice by providing more satisfactory types of denture than previously were recorded by 68% of practitioners. These positive effects of the video were greatest amongst the most recently qualified. In spite of these effects 40% of respondents indicated that they would not always carry out the whole of the denture design process, leaving some or all of it to the dental technician who does not have the clinical training necessary for the task The lack of a denture design fee within the National Health Service was identified as the main reason for this resistance to change. In the final section of the questionnaire there was little evidence for a shift of preference following the video from examples of less satisfactory designs to more satisfactory ones.

It was concluded that the programme had been well received and in the short term had produced some favourable changes in intention; more objective evidence of improvement in practice was less evident

Reference:-

Mouatt R B, Veale B and Archer K. Continuing education in the GDS - an England survey. *British Dental Journal, 1991: 170: 76-9*

Various Hypermedia Presentations Developed for Huge Amounts of Data

G. HOLWEG
Institute for Information Processing and Computer Supported New Media (IICM),
Graz University of Technology, Schieszstattgasse 4a, A-8010 Graz, Austria
E-mail: gholweg@iicm.tu-graz.ac.at

P. SAMMER, G. SORAL
Institute for HyperMedia Systems (IHM),
JOANNEUM RESEARCH, Schieszstattgasse 4a, A-8010 Graz, Austria
E-mail: psammer@iicm.tu-graz.ac.at
E-mail: gsoral@iicm.tu-graz.ac.at

We present hypermedia applications that have been developed at the IICM (Institute for Information Processing and Computer Supported New Media, Graz University of Technology) and the IHM (Institute for HyperMedia Systems, JOANNEUM RESEARCH). The demos include:

- HyperM: A PC-based hypermedia system that can handle large amounts of different kinds of data. It has been successfully used as a presentation tool at various expositions (e.g. EXPO'92 and EXPO'93) and for research projects (as for example the use of question/answer-dialogs in hypermedia systems which is especially suited for educational purposes). The project "Images of Austria" e.g. contains some 3000 high quality images with texts in six different languages, extensive maps and a number of video clips, all fully digitized.

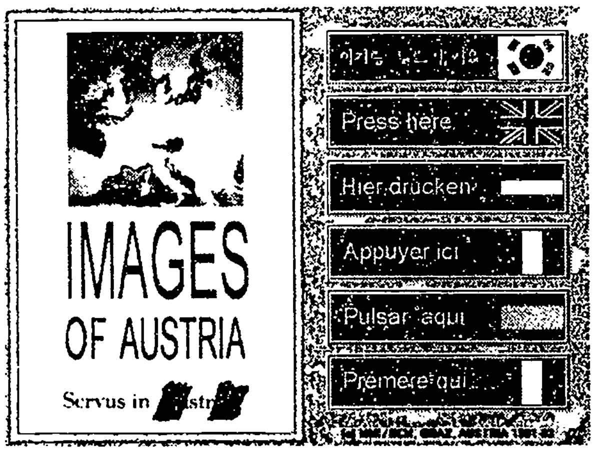


Figure 1: Start screen of "Images of Austria"

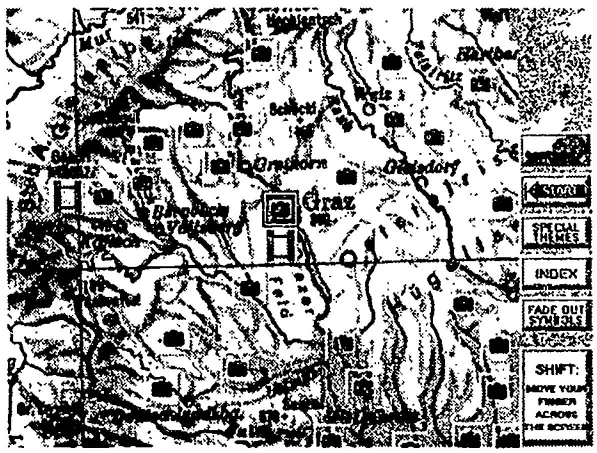


Figure 2: A part of the detail map of Austria

- Hyper-G: A workstation-based hypermedia and information system that operates in distributed networks. All kinds of multimedia information can be stored in databases on different machines. The data which can be browsed, searched, linked and annotated, can be accessed from different platforms such as workstation, PC or Macintosh.

- PC Library: An archiving system for the PC under Windows containing encyclopedias and dictionaries which pays special attention to hypertext features. Included features: keyword and full-text searches in phonetic, prefix and exact form; bookmarks, links and annotations.

Application of an Intelligent Tutoring System Authoring Shell to Develop a Course for Teaching the Fundamentals of Orbital Elements

CAROL D. HORWITZ, M.S.
Command Technologies, Inc.
USAF Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235-5352

JIMMY L FLEMING, M.A.
Intelligent Training Branch
USAF Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235-5352

Although simulation-based Intelligent Tutoring Systems (ITS) have been shown to be instructionally effective (Johnson, Fath, 1983; Lesgold, Lajoie, Bunzo, and Eggan, 1992) the cost to develop, deliver and maintain them is high. The USAF Armstrong Laboratory is seeking solutions to this cost-effectiveness problem by developing and testing a variety of ITS authoring shells. These shells reduce the cost of developing and delivering tutors by attempting to permit instructional developers and subject matter experts with little or no programming experience to develop, deliver, and maintain ITSs without relying on software programmers, knowledge engineers, or expensive hardware.

One of the first such ITS authoring shells was the Rapid ITS Development System (RIDES). RIDES was built by Behavioral Technology Laboratories at the University of Southern California for the Armstrong Laboratory. In order to test and demonstrate RIDES functionality, the Orbital Elements Tutor was built. It focused on familiarizing students with the classical set of six orbital elements, using quantitative modeling to simulate earth satellite orbits and explaining the relationship of each element to the resulting orbit. The tutor provided a test of several desired capabilities including ease of building simulations, kinds of instruction that could be generated from the simulation, and the ease with which the instruction could be modified or augmented.

References

Johnson, W.B. and Fath, J.L. (1983). *Design and initial evaluation of a mixed-fidelity courseware for maintenance training.* In *Proceedings of the 27th Annual Meeting of the Human Factors Society (pp 1017-1021).* Norfolk, VA: *Human Factors Society.*

Lesgold, A., Lajoie, S.P., Bunzo, M., and Eggan, G. (1992). *A coached practice environment for an electronics troubleshooting job.* In J. Larkin, R. Chabay and C. Sheftic (Eds.), *Computer-assisted instruction and intelligent tutoring systems: Establishing communication and collaboration.* Hillsdale, NJ: *Lawrence Erlbaum Associates.*

Hypertext As A Theoretical and Practical Tool: The TIP Project

GREG KEARSLEY
Dept. Educational Leadership, The George Washington University
Washington, DC 20052 (kearsley@gwuvm.gwu.edu)

Despite the plethora of theories and research findings about human learning, educators seldom find this information useful. This situation arises because the information is descriptive in nature whereas practitioners need prescriptions that are specific to the learning tasks at hand. In addition, learning theory and research tends to be distributed across many separate documents, making access difficult.

At a theoretical level, there is a problem of isolation between the various theories and research paradigms. Each theory and paradigm uses different constructs and terminology and hence it is hard to identify commonalities. A meta-theoretic framework is needed that makes it possible to see the overlap among theories and research paradigms as well as their unique contributions.

The purpose of the Theory Into Practice (TIP) project is to develop such a meta-theoretic framework that will also be useful to practitioners. Hypertext is used as the methodology for meta-analysis. By putting learning theories and research findings into a single hypertext database, it is possible to create links between common ideas. In addition, each theory and set of research findings can be linked to task-specific prescriptions. Thus, the database allowspractitioners to investigate specific concepts across different theories as well as locate all principles that pertain to a specific learning situation.

The specific research issues addressed in this study are: (1) What are the common and shared elements of past and current theories of learning and cognition? (2) To what extent can extant theories of learning be mapped onto a prescriptive framework for instruction? (3) How can inferences from a theory be made explicit? and (4) How useful is hypertext as a meta-analysis tool?

TIP currently exists as a HyperCard stack for the Macintosh and a Hyperties database that runs under MS-DOS/Windows. TIP is intended for use by teachers and training professionals in the design and implementation of instruction. It will also be of interest to educational researchers who are interested in hypertext as a theoretical tool.

The Hands-On Image Processing Project*

R. A. Kolvoord, M. Magisos, and D. Lahem
Lunar and Planetary Laboratory, University of Arizona, Tucson, AZ 85721

We have developed a multi-media package to introduce teachers and their students to the principles and techniques of digital image processing. This package is explicitly designed for teachers who cannot participate in a more detailed "live" workshop due to constraints of location or funding. In order to better understand the context for this package, it is necessary to look at the Image Processing for Teaching project, of which this effort is a part.

The Image Processing for Teaching (IPT) program, developed at the University of Arizona Lunar and Planetary Laboratory, is educating teachers in the use of digital image processing and in the scientific content of digital images. Recent advances in computer and imaging technologies make it possible to place professional quality research tools and materials in the hands of learners of all ages. Using these resources, master teachers in the IPT program have developed classroom activities for students in a wide range of grade and subject areas using the technology currently available in schools.

In the pilot phase of the IPT program, funded by the National Science Foundation and Apple Computer, students have been introduced to image processing in over 80 schools in 16 states, from kindergarten through high school. These schools represent a diverse range of student populations and geographical settings. Experiences at these test sites show that we have discovered an extremely effective way to excite both traditional and non-traditional learners about science and mathematics. IPT activities place the student in the role of decision-maker, using a constructivist approach to learning and encouraging the use of cooperative learning strategies in the classroom.

The IPT program is currently in its dissemination phase. Funded by the National Science Foundation, the program is conducting in-service workshops for schools nationwide through the Center for Image Processing in Education (CIPE). Curriculum development is continuing, with the production of activity units and data sets on CD-ROM. In addition to curriculum materials and resources covering a wide range of disciplines, CIPE provides follow-up support for teachers by phone and computer network.

There is considerable demand for the Image Processing for Teaching materials and workshop. A number of rural districts now have satellite receivers and sophisticated hardware. We have found this to be the case in areas as disparate as the Alaskan bush and Navajo reservation schools. Our standard workshops, with large groups of teachers from a district or a combination of local districts, cannot easily serve these rural populations. There are many schools that for various reasons (insufficient budget, size or remote location) cannot obtain the group training. There is a need for a remote education capability.

We have met this need by developing a CD-ROM-based package using the latest advances in computer-based learning, thereby allowing us to disseminate this unique program to locations and student populations that would not otherwise have access to it. This remote education course will integrate video from teacher workshops and classrooms where the program has already been implemented and which model the IPT style of open-ended inquiry and discovery, real-time interactive software demonstrations, and exercises using an integrated multi-media package centered around an interactive hypermedia program. As a teacher works through the course she/he will learn to use the NIH Image software by working through training activities, see image processing activities being developed, listen to teachers talk about IPT implementation in their own classrooms and see students as they work with the material. The course will be self-paced and open-ended, in the spirit of inquiry and discovery that is the IPT hallmark.

We will demonstrate our hypermedia package and give people an opportunity to have their own hands-on experience.

*Funded by the Annenberg/CPB Math and Science Reform Project

R-WISE
Reading and Writing In a Supportive Environment

V. LARRALDE
Command Technologies, Inc.
U.S. Air Force Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235-5352
E-Mail: Larralde@rnd.brooks.af.mil

P. CARLSON
U.S. Air Force Armstrong Laboratory, 7909 Lindbergh Drive, Brooks AFB, TX 78235-5352
E-Mail: Carlson@rnd.brooks.af.mil

Goal: The goal of this project is to design, develop, field and evaluate an intelligent tutoring system to teach critical thinking skills, as manifested in the reading and writing of expository prose.

Method: The R-WISE project uses a problem-solving paradigm to explain writing and emphasizes the power of representation in helping the writer to move gracefully through the activity spaces of the task (Neuwirth, 1989). The complete tutoring system adapts Bereiter & Scardamalia's (1987) notion of a *content space* (summarizing, analyzing, and synthesizing information about the topic) and a *rhetoric space* (planning and organizing the domain information into a logically and stylistically appropriate formal artifact). Six distinct suites of tools work in tandem to teach powerful patterns of behavior and to help with the mental overload: Crossword Puzzle, Sticky Notes, Freewriting, Cubing, Idea Board, and Revision. The R-WISE package also includes two other "free-writing" spaces called Thought Log and Writing Pad.

Prototypes: R-WISE has been in development since 1991 in an object-oriented programming environment. During the 1992-1993 academic school year, a prototype was implemented at MacArthur High School in San Antonio, Texas. In that same time period R-WISE was modified, readjusted and generally improved in response to comments and evaluations that it received during its pilot year. In the fall of 1993, the tools were shipped out to various schools across the nation for further prototype testing.

Results: Each test site is required to have both a treatment group and a non-treatment group. All students in each group take a pre-test and a post-test (at the beginning and end of the academic year, respectively) which include a reading test, a writing sample, and an OLSAT (Otis-Lennon School Ability Test). During the school year the treatment group uses all of the R-WISE tools while the non-treatment group uses only its "non-intelligent" subset of tools (Sticky Notes, Crossword Puzzle, Thought Log, and Writing Pad). Results of the writing sample for the 1992-1993 year showed a gain of approximately 7% in the treatment group. (Please note that although this figure alone is encouraging, caution should be used considering that the software was in beta version during the pilot study.)

References:

Bereiter, C. and Scardamalia, M. (1987). *The Psychology of Written Composition.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Neuwirth, C. M. & Kaufer, D. S. (1989). The role of external representations in the writing process: Implications for the design of hypertext-based writing tools. *Proceedings, Hypertext '89 Conference*, 319-341. Pittsburgh, PA: Association for Computing Machinery.

Issues of Phenomenography and Learning Using Hypertext Systems

S.K. Lau
Department of Business Systems
University of Wollongong, Northfields Avenue, Wollongong, NSW 2522, Australia

This paper discusses issues on the use of hypertext system as learning and teaching aid in relation to achieve learning as defined in phenomenography. Lawrence (1990) proposes a need of developing a tutoring software which simulates the one-to-one interaction between dealers as a means of teaching and reinforcing the bid and offer concept in financial training. The tutoring software supports classroom teaching and helps the students in consolidating their understanding of the bid and offer concepts. Bid and offer concepts should be taught and learned in relation to the context of financial markets and not separating from the complexities of money and foreign exchange markets. It was initially thought that if students can grasp the idea of selling prices in relation to daily commodities, such as apples and oranges, then they would be able to make transition to financial markets. However, experience of teaching staff highlighted a need not to separate the content (Lawrence, 1990). This is consistent with the view that how students learn about a concept should not be separated from what the content is (Prosser, 1993). With this in mind, a tutoring software with hypertext as an instructional medium was developed.

Laurillard (1988) proposes two models of knowledge, the "didactic" and "communication" models. In the didactic model, teacher has the control of the subject content and how it is taught. On the other hand, in the communication model, students are given control on what the content can be, how it is learned and teacher plays the role of facilitator. Thus hypertext system provides a more "communicative" medium of instruction. Students formulate their own conceptions and develop their own experiences about a phenomenon individually. However how do we ascertain that students' conceptions have changed as a result of learning in a hypertext environment? A good teacher takes initiative in understanding students' weaknesses and allow students to ask questions. A good teacher also provides guidance and advice appropriate to the levels of students' state of understanding. There should be no difference for learning in hypertext environment. In Laurillard's (1988) discussion, research into student misconceptions in the subject area is necessary because it helps students to be aware of their conceptions and decide what knowledge to receive. With this in mind, a phenomenographic study was carried out to identify the outcome space which show the relation of conceptual view of two-way price (Lau, 1992).

By carrying out phenomenography study before designing the system, a systematic approach is used to incorporate outcome space, which is a set of relation, in understanding a concept. The hierarchical structure of outcome space also serves as a mechanism of linking pre-defined path. The study provides a logical way to segment information and subject content according to what the students already know, what the students should learn, and what the missing conceptions are and misconceptions students have, in relation to each category of outcome space. It provides an opportunity to depict conception as a logical relation which form the basis of knowledge structuring and linkages in hypertext systems.

References

Lau, S. K., (1992), "Towards An Intelligent BOSS", *Proceedings of the First Singapore International Conference On Intelligent Systems*, 313-317

Laurillard, D., (1988), "Computers and the Emancipation of Students : Giving Control to the Learner", in P. Ramsden (ed.) *Improving Learning - New Perspective*, Kogan Page, 215-233

Lawrence, L., (1990), *An Interactive Bid/Offer Software Training System for Quoting in Financial Markets - Research Proposal*, RMIT

Prosser, M., (1993), "Phenomenography and the Principles and Practice of Learning", *Higher Education Research and Development*, 12(1), 21-31

DEMONSTRATION OF A CANCER PREVENTION HYPERBOOK

K.B. Levitan and L.M.Devore
Tidewater Consultants, Inc.
2700 S.Quincy St., Suite 400, Arlington, VA 22203

Cancer has become one of the three main chronic diseases of our time. Many forms of the disease can be prevented through life style changes that can be learned at school, at home, and at work. It is especially important to reach college-age students, who have the chance to adopt healthy lifestyles that can prevent many major cancers. With the aim of reaching both college students and general public audiences, we designed a hypermedia/multimedia system that can serve as both an instructional and reference system.

PROBLEMS AND OPPORTUNITIES

In designing and developing a Cancer Prevention Hyperbook, we confronted and resolved several key problems related to the dual purpose of the system and to its many audiences. To support classroom teaching requires a focused knowledge base that complements course materials, as well as interactive presentation techniques to engage students in acquiring and using information, and study activities to help students learn topics relevant to classes. To provide a reference tool requires a broad knowledge base with comprehensive coverage for continuous, often unplanned reference, as well as a complete index to the subjects, and interactive presentation techniques for personalization of information. Satisfying these different purposes and audiences presented many opportunities for creative design and development with multimedia/hypermedia.

APPROACH

The knowledge base is organized into five main topics that allowed us to address differences in interest and coverage required by our targeted audiences: 1) general cancer information to introduce the disease, screenings, prevention tips, early detection, diagnostic tests, cancer statistics, and treatment methods; 2) specific information on the 19 most prevalent cancers in the U.S.; 3) diet and cancer, including general guidelines to reduce cancer risk through diet, diet and exercise, and dietary asssments, as well as specific shopping, preparation and eating tips to prevent cancer; 4) tobacco and cancer, covering physical effects of tobacco use, secondhand smoke, assessment of smoking behavior, benefits of quitting, and strategies to quit smoking; and 5) cancer in the family, with information on cancer treatments and support services to help cope when the disease strikes the family.

The functional design aims to reach the largest installed base of IBM compatible PCs, and with such a goal, emphasizes graphics and animation that can run on a common PC, as well as includes small sound bites for special effects. The functional design highlights multimedia capabilities for interaction, personalization, and various modes of reinforcement including: numerous hyperwords and hypergraphics that allow users to get many or few explanations; color coded topics or chapters; numerous assessment tests for users to personalize screening and prevention; a note pad that is activated across all topics, subtopics and hyperwords/hypergraphics for personal notes and study tasks; study activities that actively engage users; hypersound to pronounce new, technical terms about cancer; graphics that reinforce common associations, practical steps, positive approach; a comprehensive index and glossary, as well as a full text search capability.

The poster demonstration is an opportunity to share our design principles as expressed in the system and get feedback from AACE colleagues.

Multi-User Shared Environments (MUSE) In Education

SUSAN MACKEY
Department of The Language Centre
University of Victoria
Victoria, B.C., Canada V8W 3P4
smack@sol.uvic.ca

Goal

To provide an interactive environment for students to acquire and practice communication skills.

Solution

Multi User Shared Environments (MUSE) are text based, virtual worlds for people to explore and adapt to their own needs and desires. During the exploration of a MUSE one has the opportunity to meet other players and talk to them in real time. Meeting with others actively encourages communication. Students connect to MUSEs from all around the world. During any one visit a student might communicate with people living in the same town or country as they live in or someone on the opposite side of the world. MUSEs encourage the growth of truly global communities.

The text based nature of the game improves reading skills. As the student wanders from place to place in the world each area is described and directions for looking at objects or moving to other areas are laid out. There are no pressures to rush when the student is wandering about or when interacting with another person. It is not difficult to explain that for one reason or another you may be a little slow in responding. People tend to be patient and helpful when they encounter newcomers. The student has time to formulate a thoughtful response before replying.

If the MUSE is being used for second language acquisition, one has the advantage of interacting with native speakers in the target culture. This summer the University of Victoria's Computer Assisted Language Learning (CALL) facility hopes to integrate some English as a Second Language (ESL) students into an existing educational MUSE. The object will be to have them create their own area based on a story they have all read. As they go through the tasks of creating characters for themselves, building homes and outdoor areas they will encounter native English speakers with whom they will communicate. Along with a multitude of English MUSEs, there are also worlds created in German and Swedish.

In this demonstration I will show, with input from participants, how to create and flesh out a persona, how to communicate with other players, how to explore existing areas and how to build an environment for others to explore.

Acknowledgements

The author gratefully acknowledges the help and encouragement of Jim Walters (Wlad) of Maricopa Community College in Phoenix, Arizona for allowing her to help at Camp MariMuse during the summer of 1993. This was done from the author's home in Victoria, BC Canada while the camp was physically located in Phoenix, Arizona.

The Courseware for Learning Technical English Using Hypermedia

Tsutomu Matsumoto
Department of Electonic Control Engineering
Kumamoto National College of Technology , Suya, Kikuchi, Kumamoto, Japan

Hideo Kiyohara
Divison of A&P
Kouzo Keikaku Enginnering Inc., Oozu, Kikuchi, Kumamoto, Japan

Introduction, Concept and Design

The Japanese Technical Colleges have suffered a bad reputation regarding their method of teaching English. It is a fact however that the industries of our country require graduates to posses linguistic abilities. Generally speaking, technical English terms are translated into 'KATAKANA' for books on technology. This tendency makes the present situation more complicated. Most students can't spare the time to master English phrases and its grammar because technical colleges have adopted special curricula emphasizing the necessity for studying technology oriented subjects. As our courseware is well organized based on computer skills and incorporates voice and movies etc., we believe that learners will find it fun to brush up their command of English, while looking at the screen of computer. To achieve this, we started to develop new courseware which realizes our aim. We regard Robotics as a sort of technology because our speciality focuses on mechatronics.

This courseware has several features: (1) Recording the learner's progres s (2) Introducing to robotic manipulator (3) Offering G.U.I.

This is implemented using HyperCard on Macintosh. The stacware consists of five parts as follows. (1) Introduction (2)Contents (3) Main Parts (4) Quiz (5) Final Test The main parts is made up of three sections. Each section has ten or twenty cards to learn fundamental robotic manipulator and each card provides English and Japanese texts, movie and voice sounds. The outstanding feature of this courseware is that users read the English texts. In this case, a movie can be a clue to full understanding.As for voice sounds, we can listen to them in the book order as shown on the card. Quizzes are planed to check whether a user's comprehension is correct or not.

Conclusion

We produced and worked out the courseware for Japanese college students to practice short-term, quick English of specialized words using Hypermedia. If we memorize Japanese pronunciation of technical terms only through 'KATAKANA', then we run a great risk that students won't be able to grasp the technical words necessary for oral communication. Applying this new courseware is worthwhile for all language learners to improve their listening abilities in the field of technology. This courseware is now improving.

References

Matumoto Tsutomu et all(1992). Developing reading and listening skills through HyperMedia : *Proceedings of FLEET II*

Acknowledgements

The authors would like to thank Mr.Akiyoshi and Mr. Hatakeyama for their contribution in the developing it.

Learning through hypermedia supported cooperative work: a case study in the field of environmental education

V. MIDORO AND R. BRIANO
Istituto Tecnologie Didattiche
CNR, via De Marini, 6, Genova, Italy

Hypermedia supported cooperative work (HSCW) is a particular instance of CSCW (Computer-Supported Cooperative Work). HSCW refers both to cooperative production based on a hypermedia development environment and to the fact that the product is a hypermedia system.

Exploring HSCW as a learning strategy has been one of the main aims of a research project in environmental education (EE), dealing with the problem of the decay of Genoa's historical centre. This research activity involved 60 seventeen-year-old students of an artistic secondary school, 8 teachers and 2 researchers.

In our approach the pedagogical design involves choosing an environmental problem, choosing a specific real environment, defining aims and content domain and linking the content domain to the curriculum. A significant understanding of an environment problem can be reached by interacting with a real environment embodying that problem. Interacting with this environment implies dealing with several tasks such as: formulating the main problem related to the specific environment; detecting the relevant components of that environment and their relationships; studying the dynamics of this environment. Summing up, interacting with a specific environment means performing field research, in which the students gather, interpret and communicate data and information related to that specific environment.

A single student working individually cannot conduct a significant study of any real environment given the complexity this involves. How then can a task that would be unfeasible for the individual be accomplished? Cooperative work can be a suitable solution to this problem as it augments the mechanical and information processing capacities of human individuals. In our case, the students were divided into 15 small groups. Each group studied a particular aspect of Piazza delle Vigne, a little square of Genoa's historical centre, in depth and got a general view of the whole system through the results of the studies carried out by all the other groups.

"Cooperative work" signifies people working together to produce a product. In our project, this product is a Hypercard system, which has been used in a demonstration in Piazza delle Vigne. Six computers were used by the students to guide people through the space and time of the square. The aim of guiding people by means of hypermedia was twofold: making the students understand the whole system (not just the single component they developed) and promoting a general awareness of the problems in that environment.

Piazza delle Vigne was modelled by representing its components and links and one or more groups were assigned with the task of developing a stack related to each component. Seven stacks were devoted to the buildings, one to the history of Genoa, and one to the socio-economic functions of the square, changing over the time. In general, a hypermedia system allows us to deal with system complexity through the encapsulation of each component in a module corresponding to a stack in Hypercard . This structure determines also the work organisation, since each stack is a work package assigned to one or more groups. The main stack linking the students' ones was provided by the researchers as a shell to be filled in.

Several studies claim that hypermedia systems are effective learning tools only when used by gifted students. This seems to hold true for hypermedia used in individual learning. However our experience suggests that when hypermedia are used in HSCW they can bring great innovation to the school environment and make learning motivating and fun.

References

Greif, I. (1988), Computer-Supported Cooperative Work: A Book of Readings, Morgan Kaufmann Publishers, San Mateo, California.

Kibby M.R. and Mayes, J.T. (1992),The learner's view of hypermedia, in Cerri e Whiting (eds.), Learning Technology in the European Communities, Kluwer Academic Publishers., Dordrecht.

Midoro, V., Briano, R. and Trentin, G. (1993) Assunzioni base del progetto LABNET, Internal Report ITD/CNR, Genova.

Riel, M. (1990), Cooperative learning across classrooms in electronic Learning Circles, Instructional Science 19, Kluwer Academic Publishers, Dordrecht.

The Use of the Macintosh as an Augmentative and Alternative Communication Aid

Mark Mizuko
Department of Communication Sciences & Disorders
University of Minnesota, Duluth, MN 55812-2496

The use of augmentative and alternative communication (AAC) aids with individuals who are unable to speak and/or write through natural modes has increased in the past two decades. AAC aids include both dedicated communication aids and computers. Dedicated communication aids are electronic aids that have been designed and manufactured specifically to be used as communication aids for speech/writing impaired individuals. Computers also can be used as communication aids when they are adapted with special hardware and software.

HyperCard can be used to create customized augmentative and alternative communication (AAC) aids with individuals who are unable to speak and/or write through natural modes. These HyperCard stacks may includes both text and visual symbols and control buttons (so the user can activate the synthesized or digitized speech). For those users, who can not activate the buttons with the mouse, they could access the buttons by programming linear or row-column scanning. Conventional input devices such as a trackball mouse or touch sensitive tablet can be used to control the scanning technique. An alternate input device such as Ke:nx (Don Johnson Developmental Equipment, 1990-91) could be used for individuals that require special switches. Another possibility is the use of a voiced activated switch.

To create an AAC aid, you start by deciding whether you will using visual symbols or pictures, or text or both. If you are creating a AAC aid with symbols or pictures, you can create them by drawing with a paint or draw program. The easiest way to include symbols or pictures is to import them from a clip art program, where a library of different pictures is available, or another HyperCard stack. Most clip art programs are organized according to different themes. If text is used, you must define and position fields in the card. Once these fields are defined and positioned in the card, text can be entered. After you have created the symbols and/or text for the card, you then define and position buttons to activate synthetic or digitized speech. Each button that is created has a button number, name and script associated with it. A button script contains instructions detailing the actions the computer is to perform when the button is activated. If you are using synthetic or digitized speech, you would program the message to be produced in the script of the button.

In order to use synthesized speech, you must have a speech driver, such as Macintalk 2.0 (Apple Computer, 1988-89), installed in your Macintosh. Macintalk is a software speech driver that produces speech on demand. After the Macintalk driver is installed in the System Folder, strings to be spoken may be delivered to Macintalk either in traditional English text or in a special phonetic alphabet. The installation consists of copying all the XCMDs associated with Macintalk with ResCopy XCMD 4.0 (Apple Computer, 1987-88 or some alternative copying program. Although the voice sounds robotic, especially compared to the digitized speech, the advantage of using Macintalk is that it does not take too much space on the floppy or hard disk.

If you are interested in using digitized speech in your AAC aid, the audio pallette within the HyperCard program is recommended. This tool enables you to record , edit , play and store sounds and speech. This application works well and the software is generally easy to use, but the digitized speech and sounds take a great amount of space on the floppy or hard disk.

In many cases, providing a AAC user with access to computers may open many other benefits, such as educational, vocational, and environmental control applications. In closing, it should be noted that computer applications for speech and/or writing impaired individuals have already begun and continue. The decreasing costs and increasing capabilities in computers should lead to the rapid exploration of this technology as a viable solution for many speech and/or writing impaired individuals.

References

Apple Computer, Inc. (1988-89). *MacIntalk 2.0* [Computer program]. Cupertino, CA.
Apple Computer, Inc. (1987-88). *ResCopy XCMD 4.0* [Computer program]. Cupertino, CA.
Don Johnson Developmental Equipment, Inc. (1992). *Ke:nx* [Computer hardware and program] Wauconda, IL.

Online tutorials even YOU can create!
Computer-based, multi-media, training slide shows

Rachel Moreland
Kansas State University Libraries, Manhattan, KS 66506-1200 USA

How can one be sure specific training is available when it is needed? One answer is to create your own computer based slide-show. Add narration and you have a special training tool that can be produced at a workstation in your library. Slide shows work especially well for relatively static information that must be repeated over and over to new or changing staff. My poster session demonstrates a slide show our library uses for NOTIS Fundamentals staff training.

The necessary equipment includes a 486 microcomputer with a soundboard, attached speakers, and a microphone. Enhancing the slide show with sound not only makes it more fun to watch, it increases the level of comprehension by engaging both the eyes and the ears of the trainees!

We used WordPerfect's PRESENTATIONS[1] software. It took me most of the summer to create my first slide show, learning the Presentations software as I went along. Once one gains some experience the production time is much less. I can do now in two weeks what it took me two months when I first started. We started out with inadequate equipment although we did not realize it at the time. I cannot over stress the importance of having adequate equipment, which for WP Presentations mean having a **486 with a minimum of 8 mg of memory** regardless of what you might read to the contrary.

One develops the information presented in the slide-show the same way any other training guide is done. I worked from the training guide we were already using for our NOTIS Fundamentals training. Thus, the goals, the objectives, the scope and the audience were already defined, freeing me to concentrate on the production aspect. I recommend this approach for the beginner.

WP Presentation software has many built-in helps, including 30-35 pre-designed backgrounds. One begins by selecting the background that will be used for all the slides. Then you're ready to start designing the slides. I did this at the computer. The outline feature is a wonderful help for the text. You type the information and the system automatically formats it. Creating appropriate graphics is more time consuming, but very important to the finished product. Remember, protect your efforts by SAVING YOUR WORK OFTEN!

The Presentation software lets you attach a text file to each slide. These files will become the narration script. If you work from an already developed training guide, creating the text files is mostly just typing. Be prepared to revise, revise and revise some more--especially when you are just getting started. I found discussing the slides with a colleague very helpful. All the slides need to be pretty well completed before you begin recording sound files.

Creat a copy of the script by printing the slides and speaker notes. Each slide will have a separate sound file. The software lets you preview the recording before naming and saving the file, thus always giving the option of doing it again. After everything is recorded, sound files must be attached to appropriate slides. The new file is very large because the sound files run from 250K to 600K each. Twenty slides without sound is approximately 350K. The same file with sound will be from 8-10 megabytes in size. Creating a run time program is the final step. Run time shows can be viewed without having to access the WP Presentations software. Run time makes the slide show very portable and, of course, reduces the cost since the expensive presentation software needs to be installed only on the creator's workstation.

We like having the ability to create quality, narrated, computer-based training with ordinary software and a minimum of cash outlay for equipment. I hope some will leave the demonstration today inspired to go home and create a slide show tailored especially for their own library.

[1]WordPerfect® Presentations, Version 2.0. ©WordPerfect Corporation 1992, Orem, Utah.

Power to the Students

Dr. Georg Miksch, Marion Neuhold
Computer Center
University of Economics and Business Administration
Augasse 2-6, A - 1090 Vienna, Austria, EUROPE

Marketing of public services

The University of Economics and Business Administration, as a public - service institution, has for the last two years sought with great success to market its computer-services. The target group is students. Through professional marketing we have reached a better utilisation level of our high investments (about 5 million US$ in the last 5 years) in networking, hardware and the devolpment of multimedia-based software. At the same time we have been able to reduce costs of pc-classrooms, energy, personnel and hardware through the application of technical innovations.

Situation of the University of Economics and Business Administration

With 25.000 students this university is Europe's biggest business-university. Computer training and internationalization are the major focuses of all student courses. This summer we will open a new training centre fitted with 250 networked high-performance computers (investment US$ 1,600.000) which will be accessible 24 hours per day. The use of computers and communication-technology is the principal educational goal, through which new work methods such as cooperative or workgroup-computing, mobile computing etc. are encouraged and trained.

PowerStore

The project PowerStore based on the idea to issue notebook computers to students on a loan-basis, thereby enabling students to maximize the use of available facilities. The number of 50 notebooks at the beginning of this project increased to 250 after one and a half years because the acceptance by students of this new service has been very high. Importantly the implementation of notebooks technology has led to the reduction of service costs, such as electricity, staff and rent by more than 95 percent. The initial hardware investment in notebooks is approximately a third of normal desktop-computers. The students pay a small fee, which is used to cover insurance, maintanance, staff-costs and a part of the reinvestment. In our opinion this project represents a revolutionary step towards a new form of computer-based education.

PowerNet

On October 28 1993, we opened the campus network, which is connected to the Internet, to all our students. Now students can use all Internet-services via the PowerNet-system from any pc-classroom, faculty-department as well as any telephone outside the university. Our goal was to attract 1.000 new student-users before the end of 1993. To achieve this aim and increase general awareness of the PowerNet-system we organized a grand opening-event, the „PowerNet-Party". As result of this presentation the target was reached within four days of the opening.

Next „Power"-Projects:PowerCard & PowerBrain

PowerBrain, a professionally organized multimedia library, with contents ranging from economics courses, through behaviour-training, to language-learning courses for diffrent computer-platforms, loadable over PowerNet and over Power-Store-docking-stations. PowerCard is the project-name for the introduction of a machine-readable student-ID-card, with additional debit-functions.

Conclusions - Why is marketing necessary?

Enormous investment in communication and computer infrastrukture must be used to an optimal level to reach the expected benefits. For the first time in its history, the University of Economics and Business Administration has begun to actively market its computer services. Up to now the marketing of two projects, PowerStore and PowerNet have been trusted to a professional advertising agency. Planning, execution and post event analysis were conducted in close cooperation with the IS-team. Both events were mainly financed through sponsorships of well-known computer companies. The marketing costs amounted to US$ 40.000,- for the first event and US$ 60.000 for the second. The success justified the costs in both cases. PowerStore and PowerNet will be continuously expanded and broadened to incorporate new services. Only through ongoing information of potential "clients", special marketing events, a high service level and reasonable prices can the optimal price/service-ratio be achieved, so that the demand for services never sinks below the offered services.

VISUAL (VIdeo-SUpported Active Learning) Resources

RODERICK I. NICOLSON, PETER J. SCOTT
Learning Research Group, Department of Psychology,
University of Sheffield, POB 603, Sheffield S10 2UR, UK.

DIANA SYDER, MARGARET FREEMAN
Speech Science Unit, University of Sheffield, Sheffield S10 2TA, UK.

Children growing up in this television age take it for granted that TV and video are an important source of information, but until recently most CAL specialists and course designers have made little use of video-based resources, owing not only to the obvious resource difficulties but also to conceptual difficulties in how to harness the power of video to the directed support available in traditional CAL packages. Although videos are excellent for creating interest, it is by no means clear that they are an effective means of fostering learning, since they tend to force a passive attitude on the learner, they sweep the watcher along inexorably giving no time for reflection, they create a high working memory load, and, unlike a book, they are well-nigh impossible to index or skim. A meta-analysis of existing interactive video materials (McNeil and Nelson, 1991) indicated that they were typically about as effective as traditional CAL, though the variability was high.

The objective of the project to be reported was the development of a methodology which exploited the recent technological development of low cost video digitisation for enhancing immediacy and motivation, while maintaining the tutorial support available from traditional CAL programs. The specific project described is the creation of such a resource using a video of a session between a speech therapist and a client with a voice disorder, winner of the Apple Computer Prize in the 1993 UK national 'Partnership Awards' competition. The poster will describe the 'Voice Disorders' resource and outline the methodology required for creating such a resource. A demonstration will also be available. Further information is available in Nicolson, Syder and Freeman (1994).

In brief, Apple's QuickTime™ conventions allow digitised video to be stored and played from the hard disc of an Apple Macintosh™ or IBM PC microcomputer. This provides an opportunity to create interactive video-based tutorial support which combines the motivational attractions of videotape material with the targeted, active learning available with CAL. The VISUAL (VIdeo SUpported Active Learning) shell has been developed for creating and using such resources. The technique required involves, first, digitisation of the video into a QuickTime 'movie'; second, analysis of the movie into individual segments which correspond to meaningful units of the transcript; third, creation of a hierarchical outline of the transcript into 'scenes', 'sequences', and 'clips', and 'segments'. Finally, tutorial questions, answers, and discussion issues may be interleaved with the script so that they may be asked at specific points. The VISUAL resource allows the user to interact with it via the hierarchical descriptive script of the contents, and to select and reorder just those parts of the video that they wish to use, thereby giving unprecedented user control over the resource.

A generic methodology has now been developed for the construction of VISUAL resources, and we are currently developing a meta-shell which allows the user to select from, and integrate a number of such resources. We believe that the VISUAL technique provides a cost-effective method for adding value to existing video resources, and that it is particularly powerful for applications involving 'learning by observation', one of the most natural of human learning capabilities, yet not easily captured until now in computer-based learning.

References

McNeil, B.J. and Nelson, K.R. (1991). A meta-analysis of interactive video instruction: a 10 year review of achievement effects. *Journal of Computer-Based Instruction*, 18, 1-6.

Nicolson, R.I., Syder, D. and Freeman, M. (1994). Construction of a VISUAL (VIdeo-SUpported Active Learning) Resource. *Computers and Education*., 22, 91-97.

Today's Level of Multimedia Development in Higher Education of Russia

A.V. OSIN
Infostudio "ECON" of the Russian Federation State Committee for Higher Education
109028 Moscow, B.Vuzovsky, 3/12, Russia

The problem of multimedia implementation into education being acute, the Russian Federation State Committee for Higher Education made a decision to organize and finance Russian scientific and technical program "Multimedia Technology" in 1991. This allows to financially support scientists' teams of different Higher Schools and to join their efforts in the frameworks of a single research program.

The Infostudio with a team of specialists in fields of both technical sciences and humanities has become the leading organization for multimedia products development.

Principles of Multimedia Titles Development

We use the technology of distributed development of multimedia titles on the basis of the original software. On the first stage the script of the product is inputted into the central data base. It includes a formal description of scenes, their parts and connections. Specialists at the audio, video working stations and that of computer graphics and animation, realize the scenes of multimedia titles on the basis of these descriptions. Then the realized elements of the product are sent into the data base. And at the final stage all the elements and scenes are assembled at the Editing Working Station. A data base can store several versions of separate scenes realization to give a script-writer an opportunity to choose one of the versions.

The main special feature of training courses and home application multimedia title developed in the Infostudio is using the game methods. The product script connects all the information units on the basis of either a single plot or its several versions. Besides the training courses have game exercises for mastering the material learned.

All these help the student to quickly adapt and the process of understanding information is more intensive. Besides such an approach arises greater interest for a title. The instrumental software developed by our specialists uses an object-oriented approach. The basic notion used in describing a scene is an animation object. It's described by a set of rules, defining reactions to the user's actions and the influence of other objects. Various methods can be used to visualize animation objects, i.e.: RLE-coded frame sequences, bitmap sprites, calculated projections of 3-D sprites. This concept makes rapid creation of interactive games possible. In fact we have an instrumental shell for creating computer games.

The Infostudio script-writers are specialists from various fields of art and culture as well as from Universities' Faculty.

The Developed Multimedia Titles

The products worked out by the Infostudio are produced either on CD-ROM or a video tape used both as a carrier of analogue live-video and of digital information. Mainly these products and training courses concern humanitarian subjects: "Business in Russia" (the Russian language for businessmen), "The Russian language for foreign school-leavers", "Cathedrals of Moscow", "The Power of planets" (home application horoscope). Universities in cooperation with the Infostudio have developed and now use in training multimedia titles on fundamental disciplines: electrical engineering, optics, theoretical mechanics, electrical dynamics and others.

Multimedia technologies in Russia Higher Education have been used not for long. But great interest they arise and today's rates of their development will make them successful in education in future.

THE USE OF OBJECT ORIENTED APPROACH TO MODELIZE AN ENVIRONMENT OF DISTANCE LEARNING

Souleymane OUMTANAGA
Département Formation Scientifique
Institut National Superieur de l'Enseignement Technique B.P. 1093 Yamoussoukro (Côte d'Ivoire) Tél. : (225) 64 05 41 Fax. : (225) 64 04 06

Laboratoire API, Université Paul Sabatier-Toulouse III France

We have been seeing for some years a growing development of computer systems. The communication network which is one of its branches does not escape this rule. Such a development has been possible thanks to some progress, made about the production of hardware as well as that of software. Among the lattest, computer networks take an important place. It also resulted from a study [Diaz 90] that there is a great shortage in the field of network engineers. Situated between computer science and telecommunication, networks constitute a theme which is judged to be too applied. It is advisable to first of all teach and promote this domain. This should first start with an effort of identification of the tackled concept in an approach system, putting an emphasis on the richness of the domain. Three steps are unavoidable. These are : • Make a particular effort in order to develop post graduate training so as to put an end to the shortage of researchers. •Develop in every scientific training a teaching of network which will be complementary to computer sciences knowledge. •Promote specialisation teaching in schools of engineers, either by teaching tools or by training in the presence of a teacher.

Our motivation to conceive an apprenticeship environment which will allow an integration of knowledge in communication network comes from this fact.

The objective of apprenticeship environment presented here is to maintain the technological training in the context of computer networks. This can be achieved either locally or remotely in the case of teleteaching It aims at students, also to teachers wishing to create and send their courses to learners of training centres in the field of computer networks. Most of network books available now on the market are too theoretical. So a study for more practicability allowed us to set a system of teaching of networks.

For this objective we have chosen to use an Object Oriented Programming for several fundamental reasons : -The paradigm of object is particularly adapted to distributed systems because objects are entities which contain data, methods to handle data, and attributes (properties) and these objects can communicate by messages. -The object oriented approach is also the basis of powerful User Interface Management System. -The object oriented approach is well adapted to multimedia project at user's level. From these consideration we have chosen the [ISO 10165] as ideal platform. The modelling of the environment entities is based on the object oriented approach. The preceding defined model and object formalism [ISO 10165] will allow us to build pedagogical objects (lesson, module, apprenticeship units, didactic objects, exercises, glossary, bibliography).

References

[Diaz,90]: Diaz M.: "La recherche et les réseaux informatiques", *Revue TSI* Paris 1990.
[ISO 10165]· ISO/IEC DIS 10165-2, Information Technology-Open Systems Interconnection-Management Information Services - Structure of management information.

Campus Support for Multimedia Information Retrieval: A User Interface Configuration

Charles B. Owen, Fillia Makedon
Dartmouth Experimental Visualization Laboratory (Dev Lab)
and Department of Mathematics and Computer Science
Dartmouth College, Hanover, NH 03755

Multimedia and digital resources have enormous potential for enhancing the learning process in the University of the Twenty-First century. Given rapid access to large quantities of information using advanced search methodologies, a student spends less time hunting and more time learning. This poster presentation and accompanying paper illustrate a viable architecture for such a large scale information retrieval system.

A Multimedia capable information retrieval system must include a repository for large quantities of information combined with mechanisms for searching and delivering this information to end users. This information may be locally stored, or it may be public and private information available via national networks. The data involved includes digital video clips, reference volumes, image data, sound and voice recordings, scientific data, and private information services [1]. Access to this information must be equitable across the entire campus population, including students, faculty, and staff. Since the user base is very diverse, encompassing both instruction and research, the available datasets are also necessarily diverse. A practical system must adapt to changing user, data, and equipment needs.

In this presentation, we describe and analyze the components of an architecture for a campus wide information retrieval system with emphasis on the central depository. We argue that such a system is not merely an expansion of the campus networks of today, but a vast and powerful repository of diverse types of information that can be accessed at high speed by a large number of users. It is important, therefore, to carefully plan for and design a system and interface that will anticipate the new types of data that will be available in the future, rather than simply networking large numbers of general purpose computers. This paper addresses issues which should be considered in such a design, including volume information delivery, adaptability, redundancy, storage backup and scalability.

Large Scale Information Retrieval: An Architectural Model

The architectural model presented can be broken into several major components: end user equipment, the multimedia network, network routers, data storage nodes, and central depository nodes. In order to maximize immediate utilization, the model is designed to integrate into existing computer networks and, therefore, attaches to those networks and coexists with existing storage. The major new components described in detail in the presentation are the new links in the multimedia network, new network routers, and the central depository nodes.

The presentation focuses on the Central Depository Node, as this is the common throughput bottleneck of traditional approaches to high volume information systems. The purpose of the central depository node is to provide a very large central storage point that can be accessed by a large number of users. It is argued that a large, scalable system can not be effectively built from general purpose hardware. Even using very fast general purpose hardware combined with an optimized design for multimedia server applications, the StarWorks video server can only support twenty users simultaneously [3]. Hence, our modular Central Depository Node design is built from storage nodes, stream channel controllers, and network interfaces. Additional components, including knowledge processors and a master catalog are integrated into the design. The model allows for evolutionary growth as the number of users and the volume of data increases.

A storage node consists of storage devices and control. The stream channel controllers provide buffering to manage the delay sensitive delivery of the multimedia data streams [2]. An important proposal of the model is the isolation of the stream channel controllers from the storage nodes. The stream channel controllers are thus shown to be usable for not only buffering, but synchronization and object composition.

References

[1] Garrett, J. R.. Digital Libraries, The Grand Challenges. *Educom Review*, 28(4) (Aug. 1993), 17-21.
[2] Gemmell, J. and S. Christodoulakis. Principles of Delay-Sensitive Multimedia Data Storage and Retrieval. *ACM Transactions on Information Systems*, 10(1) (January, 1992), 51-90.
[3] Tobagi, F. A. and J. Pang. StarWorks -- A Video Applications Server. In *Digest of Papers: IEEE COMPCON Spring '93*, 1993, 4-11.

Mesa Verde National Park, Mesa Verde Museum Association, The Avalon Group and Kansas State University
An Interactive Collaboration

John D. Parmley
Associate Professor
Department of Secondary Education

Robert F. Hower
Associate Professor
Department of Art

Kansas State University, Manhattan, Kansas, U.S.A. 66506

The development of interactive materials to be used in schools or homes across the United States or in international settings as well as within the host Park site, is the major focus of a partnership which involves Mesa Verde National Park, the Mesa Verde Museum Association, an interactive multimedia development company known as The Avalon Group and Kansas State University. The partners have entered into this formal collaborative arrangement to enhance the educational opportunities of the clientele served by schools as well as the clientele served by the Park. The products being designed by the partnership are interactive and take advantage of a variety of media. The first product is currently under development and is designed for students in school settings, but may also be used by visitors and others with an interest in the cultural history of the area. It will feature an instructional CD, a resource or "scrapbook" CD, a teacher's manual, learners' guide and an excavation simulation project. This product should be ready for distribution before September 1994.

Mesa Verde National Park as the first United States National Park dedicated to the preservation of a cultural heritage, offers significant opportunities for visitors of all ages to explore a wide variety of issues and problems that impacted a prehistoric Native American culture known as the Anasazi culture. This culture developed in the American Southwest from about the time of Christ to approximately 1300 A.D. The use of interactive computer technology, such as those incorporated in the partnership's first product, provides opportunities for student users as well as visitors and interested individuals in distant locations who are not able to visit the Park, to explore the cultural and natural resources of the area with a degree of specificity often not available to traditional on-site visitors. Such interactive products are seen as very beneficial in helping the Park meet the needs and interests of many visitors including those who are physically challenged and unable to traverse the often steep terrain.

The special talents combined within the partnership provide opportunities for teacher educators and curriculum material developers to interact with graphic artists, dynamic media specialists, technology specialists and the Park's archeological and interpretative specialists. Kansas State's College of Education faculty bring effective teaching talents, related experiences in developing curriculum materials and talents in the use of instructional technology to the collaborative team. At the same time, since many cultural practices are difficult to recreate and many locations are extremely difficult to access, the involvement of the Kansas State University Department of Art is especially important. Graphic artists and dynamic media specialists have been asked to conceptualize and animate numerous concepts including working with members of The Avalon Group to develop the overall aesthetic quality of each product, thus enhancing the effectiveness of the product.

Initial conceptualization of a project involved representatives of the four participating groups developing the framework from which each group will contribute their expertise. The Avalon Group provides leadership for the development of a storyline and presentation approach employed within the product. Avalon Group staff members work with the Park Service to gain access to remote or restricted locations to develop many of the video and still image resources needed for the product. KSU teacher educators work with the Park staff to provide leadership for identifying the educationally significant aspects of various sites and the Anasazi culture in general as it evolved over time. Teacher educators then develop curriculum materials to be included in the teacher's manual and student guide. The Art Department is providing the leadership to develop the resource of "scrapbook" CD to be used by students as they develop their interpretation of the Anasazi culture. Finally, all members of the partnership interact with the Park Service staff to establish the authenticity of the product. Finally, The Museum Association is providing additional consultation on the look and feel of the product as well as input into the development of marketing strategies.

A Computer-based Interactive Physical Education Resource

B. PASTERIS
President
INSEGNA INC., 103 Bracebridge Cr. S.W., Calgary, AB, Canada T2W 0Y6

As class sizes are increasing, it is difficult for students to receive the amount of one on one instruction they desire and need. Normally, the teacher demonstrates a skill in front of a group of students and then gives instructions on how to practice it. As much of the instructors time is spent answering 'routine' questions, the teacher rarely has enough time to help individual students with skill analysis.

Sports Software

To overcome these difficulties, INSEGNA has developed an interactive multimedia sports program which contains all the skills, strategies and rules for a particular sport. Our programs are organized in such a way, that the learner is able to progress from skill to skill following the correct scope and sequence; or can choose to focus on a specific skill, rule or strategy.

Our programs feature the best of multimedia technology; animation, digitized video, and sound. These elements are used to instruct learners on correct skill technique as well as to test their knowledge of correct skill performance, game rules and game strategies. Complete, detailed instructions on correct skill technique and lists of performance cues are included. Both the performance cues and detailed instructions can be printed out and presented to students for study purposes once they leave the computer.

Other key features of our programs include a list of appropriate drills, and colour skill coding to differentiate beginner, intermediate and advanced skills.

The Advantages

INSEGNA's sports software is a valuable resource for students and teachers alike. It functions as a teaching tool, an instructional tool, a remedial tool and it allow for individualized learning. Students who have already mastered a skill can return to the program and receive additional instruction in new skill areas, as well as related drills, game strategies, and game rules. Whereas students who are having difficulty performing a skill correctly, can receive additional drills and reminders on correct technique. Our programs also free teachers so they are able to spend more time on skill analysis, error detection and error correction.

Finally, our sports software makes lesson planning much simpler, as all of the required resources are provided in our packages. Consequently, teachers with little sports related knowledge can provide instruction which is pedagogically correct, effective and relevant.

Implementing *Hyper on the Library*™: Computer-Assisted Library Instruction on HyperCard®

ELLEN J. PERLOW
B. Davis Schwartz Memorial Library
C.W. Post Campus, Long Island University, Brookville, NY 11548, USA

Hypertext-based interactive computer-assisted instruction [CAI] programs increasingly are being utilized as alternatives to traditional instruction models. Whether they produce as high a level of conceptual learning as the traditional methods or are "a superior way to learn" (Bourne, 1990) is still an open-ended question. In any case, evaluation of their effectiveness is required to properly judge when, where and how they should be adopted.

One critical variable is whether the program is used as a supplement to or a stand-alone replacement for classroom instruction. Reported research results suggest that differences in post instruction performance may indeed be small. However, a combination of classroom and cost-effective hypermedia instruction may offer the best solution in terms of user performance (van-den-Berg & Watt, 1991).

Hyper on the Library™, CAI on HyperCard® (demonstrated at ED-MEDIA 93 at a pre-testing stage), was developed by the author for the University's expanded library competency curriculum. During the 1993-94 academic year, it was piloted in independent study and laboratory class settings and compared to traditional classroom instruction, in anticipation of its implementation as a course option in the Fall 1994 semester. Like another such program (Mackey, Dugan, Garrett, & Freeman, 1992), its content and basic structure parallels those of hard copy manuals employed in other course options. From observation, analysis of participating student evaluations and consultation with colleagues, the program's instructional design, whose importance cannot be underestimated (Grabinger, 1993; Okolo, Bahr & Rieth,1993), has been significantly revised, enhanced, simplified and improved. The program's positive reception by students with varied prior computer experience substantiates the advantages often attributed to hypermedia systems, in particular, HyperCard®: flexibility, user friendly interface, multimedia audio, visual, and now color capabilities, and nonlinear user discretion for content, time, place, and pace of instruction (Okolo, Bahr & Rieth, 1993).

However, the lack of any imposed structure also may be a disadvantage. The student can become disoriented within the nonlinear program, and can suffer from information or "cognitive" overload (Conklin, 1987). To avoid this pitfall, *Hyper on the Library*™ is divided into independent learning units or "conceptual neighborhoods" (Spoehr & Shapiro, 1991), corresponding to the Library's various departments. In addition, a set of preprinted worksheets are completed while navigating through the computer units. Answers to the questions, found in sequence on program screens, assist students in completing accompanying hands-on exercises in Library departments (common to all course options). Though certain nonlinear freedom is lost, using computer-based and print media within an imposed yet still flexible structure has been found to be beneficial both to the learning process and to program management, evaluation and revision.

References

Bourne, D.E. (1990). Computer-assisted instruction, learning theory, and hypermedia: an associative linkage. *Research Strategies*, 8 (4), 160-171.

Grabinger, R.S. (1993). Computer screen designs: viewer judgments. *Educational Technology Research & Development*, 41 (2), 35-73.

Mackey, N., Dugan, C., Garrett, W., & Freeman, C.L. (1992). Teaching with HyperCard in place of a textbook. *Computers in Libraries*, 12 (9), 22, 24-26.

Okolo, C.M., Bahr, C.M., & Rieth, H.J. (1993). A retrospective view of computer-based instruction. *Journal of Special Education Technology*, 12 (1), 1-27.

Acknowledgements

The author acknowledges the support of grants from Long Island University's C.W. Post Research Committee and the assistance of Dr. Peter Liebscher, Palmer Library School, Long Island University.

DEMON : A Distant Educational Monitoring Project

VALERY A. PETRUSHIN, VLADIMIR I. GRITSENKO, ALEXEY M. DOVGIALLO
Glushkov Institute for Cybernetics, 40 Prospekt Akademika Glushkova, 252650 Kiev, Ukraine
E-mail: petr%itslab.kiev.ua@fuug.fi

Today in Ukraine there are enough number of computers which are connected to distributed networks. From five worldwide computer networks (BITNET, INTERNET, UUCP, FIDONET, OSI) two networks UUCP (RELCOM) and FIDONET are widespread supported in Ukraine. Last year we have obtained a direct connection to INTERNET. Mostly these networks use for supporting e-mail, for commercial and research applications. There are also several BBSs for software exchange. There are no applications for educational purposes but this kind of applications is the most prevailing in Western countries using INTERNET network (Krol, 1992). However we had some experience of educational networking in Ukraine because we have here about 150 schools which were chosen for participation in the IBM program "Pilot schools" for the USSR. These schools took part in HELLO and ACID RAINS projects which included exchange of letters and data by e-mail (through UUCP) between Soviet and American participants. At present a project for creating and providing the Educational System Integrated Computer Network is under development.

Last year at the Glushkov Institute for Cybernetics the DEMON (Distant Educational MONitoring) project was started for solving software and pedagogical aspects of the third problem. Since "monitoring" means a process for supporting something in stable state then "distant educational monitoring" means the process for supporting knowledge of educational administrators, teachers and students about current state of educational systems in Ukraine and other countries, teaching methods, curricula, learning materials on various media including computer tutoring systems, and etc. on the base of distributed computer networks. We consider the DEMON project has many goals in common with the HYPER-G (Kappe, F., Maurer, H., Sherbakov, N.,1993), and GOPHER and W^3 are the samples of successful design of user interfaces.

The main goals of the DEMON project are following:

- Creating software and courseware for distant learning.
- Developing structure of databases and teleconferences on teaching methods, curricula, learning materials, and etc. for various subjects and levels of education.
- Developing software for creating, modifying, and maintaining distributed educational databases.
- Developing the technology for creating computer-based tutoring systems.
- Developing techniques for using computer networks for distributed pedagogical experiments and expertise.

We consider educational administrators, teachers, and students as three categories of users. The administrators' activities using DEMON will include: searching and maintaining of relevant legal documents, rules and regulations; meeting organization and maintenance of minutes; room management and timetabling; maintenance of student records, budget reports, etc. The dominant teachers' activities will be: creating and distributing courseware and other learning materials; discussing questions of common interest through teleconferences; answering students' question by e-mail; taking part in distributed expertise of learning materials. The students' activities will be: searching for relevant learning materials; browsing through relevant documents; choosing, fetching and executing courseware; asking questions e-mail and teleconferences. In the future we consider to add other services such as the access to telephone directories (white and yellow pages), e-mail address directories, encyclopedia, specializes handbooks, and bi- and multilingual dictionaries.

References

Kappe, F.,Maurer, H., Sherbakov, N.(1993) Hyper-G: An Universal Hypermedia System. *Journal of Educational Multimedia and Hypermedia*, 1993, 2(1),39-66.

Krol, Ed (1992) The Whole INTERNET. User's guide & catalog. O'Reilly & Associates Inc., 1992.

An Evaluation of Computer-assisted Learning Programs for Training General Dental Practitioners in Denture Design

DAVID POLLARD AND JOHN DAVENPORT.

The School of Dentistry, University of Birmingham, St Chad's Queensway, Birmingham, B4 6NN, UK.

Computer-assisted learning (CAL) can combine all of the commonly used modes of presentation (concrete, symbolic and visual), it can promote deep, self-paced, active learning and can accommodate both a problem-solving holistic, and the more fact-oriented serialist, approaches to learning.[1] Dental practices in the UK are increasingly becoming computerised and it is the policy of the Dental Practice Board that the majority of practices will be computerised within the next few years. The Dept. of Health has made a grant available to cover part of the cost of installing computer systems and, as a result the availability of hardware which can support dental CAL is increasing steadily.

The objective of the present project was to evaluate CAL programs on aspects of denture design as a distance-learning package for dentists. Four CAL modules were developed from chapters in a textbook on the subject by one of the authors[2]. The programs were developed using the Toolbook programming language for PCs using Windows version 3.0 with 2 MB of RAM, a 386 processor, a colour VGA screen and a mouse. Particular emphasis is placed on rich forms of interaction creating a constant dialogue which includes the direct manipulation of on-screen graphics. Action by the user results in feedback to explain the basis for the correct answer. Three consecutive incorrect attempts bring up a help button which gives the correct answer if required. An index allows the user to navigate easily through the program. A glossary can be accessed by clicking on any dental term with the right button of the mouse. On completing the program the user is given a performance rating.

Sixty-five dentists were recruited as evaluators. Those with suitable computers were sent the programs for use at home, while the others used the programs at the dental school. All participants completed a questionnaire consisting of fixed and open questions with the former requiring answers on a scale of 1-5 where 1 was the most negative response and 5 the most positive. The costs of this evaluation were met by a grant from the Department of Health. The results indicated that modules took 10-45 minutes each to complete. The positive results of the survey were: 85% found the programs easy to use; 78% reported that the programs extended their knowledge of the subject; the programs were considered to be more useful than videos (66%), audio tapes (83%), journals (74%) and books (65%); 80% were interested in using other CAL programs, 64% would consider buying other CAL packages (35% @ £20, 47% @ £50, 18% @> £50). The commonest positive open comments related to the clear graphics and to the use of animation. Negative responses were that only 43% considered that the programs extended their practical skills (however, this was not an objective of the programs); 30% were frustrated by invisible target zones being too precise; a few suggested that the clarity of some of the diagrams and text could be improved.

The programs were modified extensively in the light of these comments and 60 copies of the revised versions have now been purchased by the Department of Health and distributed, with appropriate hardware, to postgraduate centres throughout the UK.

References:-

1. Teaching and learning in an expanding higher education system. *Report of a working party of the Scottish University Principles. SCFC, 1992.*

2. Davenport J C, Basker R M, Ralph J P and Heath J R. Atlas of Removable Partial Dentures. *Wolfe Medical Publication, London, 1988.*

The Bell High School Video Portfolios CD ROM

BRIAN REILLY
Division of Language and Literacy, School of Education
University of California, Berkeley, Berkeley, CA 94720, USA

Overview

The Bell High School Video Portfolios CD ROM is an interactive electronic portfolio containing QuickTime™ versions of 28 videos produced by students in the Bell High School Television Production program. Each video is accompanied by an interview with a student producer, along with comments on the video by the television production teachers, Ed Murphy and Larry Stone. The videos represent the range of productions done by Bell students, including Public Service Announcements, Video Essays, Video Poems, and Music Videos.

This CD ROM is unique — it collects the work of students into a group portfolio, giving the reader/viewer an opportunity to learn about the award–winning Bell High School Television Production program by seeing videos, watching and listening as students describe their work, reading comments by their teachers, and seeing student work in the context of similar projects by other students. The interface provides a number of ways (genre, title, author, interviews, video index) to access the more than 1 hour and 50 minutes of Quicktime video on the disc.

Background

Bell High School has a very successful video production program, certainly the leader in the Los Angeles area and probably throughout California. Bell students won twenty-three different awards during the '92-'93 school year for their videos in local, state, and national competitions, and they've won about the same number of awards each of the previous two years.

The success of the students in Bell High School's television production program is even more noteworthy given the context in which they work. Their school has 4200 students and due to crowded conditions, has operated on a year-round schedule, with three tracks for the past 13 years. Many classes are over-enrolled, with more students than available chairs. Like many urban schools in California, Bell High has a 40% dropout rate, and is not far from gang and drug activity. The Bell video production facility has one editing set-up with an Amiga Video Toaster, three or four working camcorders, and four Apple IIc's for scriptwriting. Despite these conditions, Bell students continue to produce a range of high-quality videos that have brought themselves and their program considerable recognition.

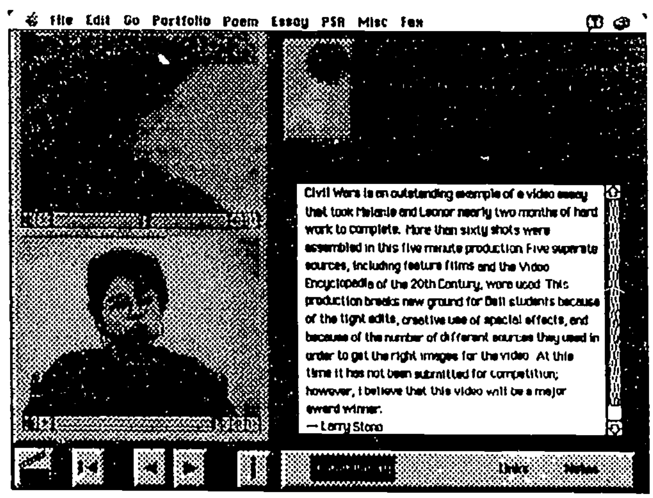


Figure 1 – Screen shot from Video Portfolios CD ROM showing entry for "Civil Wars," video essay by Melanie Alvarenga and Leonor Martinez.

Electronic Groups Communication via Satellite as New Distance Learning Environment

Dr.sc. FJODOR RUZIC
Institute for Informatics, Markusevacka 7, Zagreb, Croatia

The success of information technology utilization in the distance learning environment is strongly dependent on the communications media/platform. For the population of learners in the distance learning environment, there is a need for the development of communications networks upgrading them from leased or switched telephone lines to wireless, satellite based communications networks. Satellite networked communications platform and information bases are the tools for creating new independent, and open educational environment. Open learning environment has the idea that education is important not only because it contributes to one's career goals but also because of the value it adds to the general quality of one's life. Schools in distance learning environment develop networks that link any number of computer/desktops at learners' and teachers' sites, and let them share programs, files and messages. Educational institutions are acquiring networks for the same reasons business are - in increasing productivity and the quality. Interactive technology needed for computer supported collaborative work in distance learning environment can assist students inquiring on their own and that frees the teachers to work on specific teaching/learning problem. Many school libraries tap into on-line services, and provide access to thousands of university databases through Internet (through CarNet in Croatia specifically). When students can reach all that from their desks, the teacher's role changes from delivering information to helping students understand it. In the same time, computer and communications technology make the room for new methods of learning that replace detailed curricula with multidisciplinary projects and emphasize teamwork. This is why computer supported workgroup (electronic group) and communications networks are needed for today's learning environment.

It is available today for any type of digital communications network to be established in VSAT environment. The cost of satellite airtime and space segments as well as the cost for earth stations therefore depend entirely on the functionality of the network. In the same time, various data rate requirements and bandwidth options offer different cost-benefit performances, thus making a really trade-off between the cost of the earth stations, the total number of earth station/VSAT units envisaged in a network, and the recurring cost of the satellite airtime/space segment based on the degree of the providing satellite power (in spot or other means of beams) and bandwidth required to establish an operational network for dedicated distributed applications environment. The currently available 22/30 GHz frequencies range offers the opportunity to achieve relatively high bit rates with very compact ground signal status rearranging VSATs to be capable of transmitting and receiving information thus providing interactive medium. Given the identical antenna sizes of VSAT stations, the high frequencies result in higher antenna gains as compared with the 12/14 GHz frequency range mostly in use across Europe. This is attractive momentum that benefits applications in which learners and teachers are dominantly oriented toward interactive communications services at low costs. The implementation of an interactive VSAT system distribution network should be aimed to users' environment and should provide flexible and variable organization of distance learning; it must be open to various types of contexts in which it can be used. The organization of telecourses made by VSAT network opens up three main activities needed to create educational environment:

- creating of central lecturer function with adequate administrative and technical installations
- creating of central information (multimedia) base reachable by any point connected to VSAT network
- implementing minimal technical (software and hardware) and instructional elements

This is only general plan that is proposal for one year task schedule including the parallel activity considering the selection of the space segment operator (that will be EUTELSAT for the first stage), the selection of the Hub equipments and terminals (stand-alone earth stations/transceivers). All of them lay in the field of technical solutions that are known recently. The problem to be solved is to find right organizational form that accommodates both the existed formal school system and the acute requirements found in the informal education. VSAT systems improve the communications facilities only, so I found that organization of distance learning environment should be done prior any decision for startup VSATs for educational purposes.

A Conceptual Framework for Building Hyperstories

J. Sánchez and A. Mallegas*; L. Cernuzzi**; G. Rossi, M. Lumbreras, A. Diaz and L.M. Bibbo***
* Universidad de Antofagasta, Antofagasta, Chile; ** Universidad Catolica de Asunción, Asunción, Paraguay;
*** Universidad Nacional de La Plata, La Plata, Argentina

We present a modern view of producing multimedia software for learning. This study was developed to enhance thinking and learning through the use of hyperstories (Ref. 1). Stories are narratives of true or fiction events that intend to capturate and involve actively the learner. Based on the static metaphor of a typical iberoamerican neighbourhood ("el barrio"), representing a place were children live and develop socially and culturally, we have designed a dynamic metaphor we call hyperstories. "El barrio" is a metaphor where we can find houses, buildings, parks, schools, churches, cultural centers, etc. Interesting stories occur in "El barrio" involving and motivating the learner, as well as giving him/her control over the stories, tools and construction materials to build things and develop strategies to test hypotesis with the implicit idea of fostering the development of cognitive structures that determine tempo-spatial and laterality relationships among children ages 6-8. Hyperstories are, in some way, the electronic version of conventional literacy stories in the same way that hypertext is the electronic version of text. However, we push forward the metaphor in order to allow a "dynamic binding" between characters, the world in which they move and the objects they act on. This binding is performed by the learner thus allowing greater flexibility in the learning process. Thus, a hyperstory is the combination of a virtual world where the learner can "navigate", a set of objects on which the user can perform certain operations, and a set of characters that can be manipulated by the learner. Objects and characters may have their own behavior and act autonomously.

A conceptual model for hyperstories

Conventional hypermedia authoring tools does not provide an adequate set of facilities for building hyperstories. The static environments involved in a hyperstory (the virtual world) can be simulated easily, but several aspects such as dynamic objects behaviors and complex interactions between the main character and others exceed the conventional "nodes and links" model. By using a specially defined a conceptual model for hyperstories we can describe the virtual world as a nested context model (Ref. 2). The world can be enriched with objects and characters and thus "instantiating" a particular story. A virtual world is defined as a set of contexts. Each context contains an internal state, a set of contained contexts, a set of objects, and links to other contexts. Different relationships may be held between two different contexts. Different "real world" metaphors can be implemented easily with this simple model. Moreover, contexts may be reused in different virtual worlds.

The main difference between our model and traditional hypermedia models (Ref. 3) is that nodes (contexts) may be nested. However when we add objects, the world is enriched. State variables and a behavior describe the attributes of a certain object. The object's behavior is specified by using rule-based scripts. Each rule contains a pre-condition and a list of actions that must be performed when the pre-condition holds. Objects are further classified into statics and dynamics. A static objects always belong to the same context and dynamic objects may be carried from one context to another. Note that certain contexts, for instance books, can be also carried from one container context to other. We call them dynamic contexts.

In addition, objects can perform discrete or continuous activities. A door can be opened or closed (discrete) and a recipient may get full of water (continuous). Finally, one of the most interesting characteristics of certain objects is the capability to represent links between contexts. When manipulating characters the learner acts upon the world, navigates, performs actions on objects, carries objects from one context to another, etc.

Considering the aforementioned, a hyperstory is then the combination among contexts, objects, characters, and the interaction patterns performed by the learner. Events produced by the learner while interacting with the virtual world, the time sequences, and the interaction among objects determine how a particular "instantiation" of a hyperstory will look like.

References

1) J. Sánchez. The conceptualization of hyperstories. Working Document., 1992

2) M. Casanova et al: "The nested contex model for hyperdocuments". Proceedings of Hypertext'91, 1991.

3) F. Garzotto et al.: "HDM. A model for the design of Hypertext applications". Proceedings of Hypertext'91, 1991.

Creating CAL resources using PsyCLE: the Psychology Computer-based Learning Environment

PETER J . SCOTT
Learning Research Group, Department of Psychology,
University of Sheffield, POB 603, Sheffield S10 2UR, UK.

NICK HAMMOND, JEAN MCKENDREE
Department of Psychology, University of York. York YO1 5DD, UK.

The PsyCLE Project is a three-year large consortium project involving over 30 Universities to develop computer-based material to support introductory University level teaching in Psychology. The project aims to deliver significant modules in 10 areas of the subject on both PC and Macintosh microcomputer platforms. The initial direction of the project involves the development of a suite of authoring tools and templates based in HyperCard on the Macintosh to support the 10 development teams. The current prototype PsyCLE development platform is focused upon a concept-mapping tool which allows the lecturer author and the student to construct node and link maps of information.

In August 1992, the UK Higher Education Funding Council announced a call for proposals for an ambitious plan to promote use of computer technology in higher education called the Teaching and Learning Technology Programme (TLTP). The ultimate object of this Programme "is a decisive extension in the use of technology in learning so that, where appropriate, it becomes an integral and established feature of the delivery of higher education". A major focus is on the efficiency of the educational process whilst nevertheless maintaining effectiveness through the potential benefits of improved quality of courseware and flexibility for students.

The Universities of York and Sheffield are the lead sites for the Psychology TLTP consortium. We are working on a courseware design and delivery platform known as the Psychology Computer-based Learning Environment. We have nine major sub-discipline development sites at Universities in the UK. Each is responsible for a particular sub-discipline based module. These ten modules are: Auditory Perception (Bristol), Visual Perception (Birmingham), Developmental (Warwick), Social (Kent), Statistics (De Montfort), Experimental Design (Reading), Psycholinguistics and Neuropsychology (Manchester), Collaborative Group Work Support (Nottingham), and Philosophical Roots (Middlesex). Each development site is matched with some of over 23 evaluation sites in the UK and 8 overseas sites in the USA, Australia, Denmark, Sweden, and New Zealand. We are developing core material plus additional topics in each of the modules, whilst also providing a means for instructors to modify content, sequencing and associated learning tasks to suit the needs of their particular courses. For instance, our evaluators teach a range of students from different colleges. They may not want to show the same material to nurses as to honours psychology classes.

Our emphasis is on interaction and engagement with multimedia materials rather than primarily on presenting textual and graphical information, as in many current CD-ROM and other multimedia products. We are concentrating in many cases on demonstrations and laboratory experiments which are hard to convey in conventional teaching and which can effectively exploit the power of the computer to make more informative and advanced experiments possible. For instance, the student can work with the experimental design module guiding them to run an experiment numerous times, but with different population samples to see the effect of sampling size and characteristics. In the developmental module, the students make predictions about children's behaviour according to various theories and then will be able to compare their predictions to video clips of children performing the tasks.

Reference

Hammond, N., McKendree, J., Scott, P. J. and Trapp, A. (1993). Resources for teaching introductory psychology: the psychology computer based learning environment. *Psychology Teaching Rev.*, 2 (2), 71-80.

Towards a multimedia and multimodal courseware architecture

Ahmed SEFFAH and Kamel OUADOU
MIS-LISPI ECOLE CENTRALE DE LYON
B.P 163, 69 131 ECULLY CEDEX
Phone : 72.18.64.41 Fax : 78.33.16.15
Mail : seffah@cc.ec-lyon.fr

A multimedia courseware enables the manipulation of multimedia information (texts, sounds, graphics, bitmap or video images). A multimodal courseware enables the analysis and fusion of events from the keyboard, the mouse or the microphone devices. A multimedia architecture needs technics and tools to organize and browse through the multimedia information. Morever, the multimodal architecture needs a protocol to generate multimodal events from the simple input events but also to analyse them in the learning situation.

In the domain of interactive application design, searchers have already proposed and ratified several architectures [Kazman 93]. These architectures are absolutely transposable to C.A.I insofar as a courseware is also a high interactive application. The architecture that we propose relies on the integration of specific components such as the Intelligent Tutoring System (ITS) in the Seeheim architecture [Pfaff 85]. As the Seeheim model does our architecture distinguishes three layers :

- **application** is composed by a set of semantic functions which represents the courseware functionnal core. For example a courseware on Computer Assisted Design (CAD) concepts needs processing to manipulate curves and surfaces.
- **control** is based on two components i.e. the training adviser and the learning manager. The adviser task is to put the learner into the most suitable learning units. The learning manager controls the temporal execution of the learning units and captures the learner cognitive behaviour. The adviser analyses the behaviour parameters captured and generates a new learning situation.
- **presentation** distinguishes several components i.e. the presentation manager, the monomodal events managers and the multimodal generator. The presentation manager takes care of the restoration of the multimedia information on the output devices. It selects the information to present by browsing through the multimedia database. The monomodal events managers control the end user interactions. Each one generates a monomodal event and put it in the monomodal events queue. This queue is used by the multimodal generator to produce multimodal events. The generator continuously searches command events in the queue and merges all monomodal events which have relation with the command event. The fusion method uses the general model of interaction <Command-Name, Objects, Parameters>. The generator seeks the value of each parameter of the requested command. The multimodal event produced is directly sent to the learning manager which establishes an association between the event and a semantic function.

Our architecture distinguishes two kinds of components. The specific components of a courseware such as the learner profile, the learning units or the semantic functions units. The common components of all coursewares (the generic components) such as the learner manager, the training adviser, the presentation manager, the multimedia generator and the monomodal events managers components. These components can be implemented once and for all and reused to develop all the coursewares.

As an experimentation, we implemented two releases of the generic components. One is developped under the Ms-Windows system with the Turbo Pascal and the C++ programming languages. For the second we used the Macintosh Hypercard system.
We reuse the generic components implementations to build two coursewares : MATH-MAX and EAO-CAO. They are respectively dealing with the basic arithmetic learning and the CAD concepts. They respectively use the hypercard and the Ms-Windows releases.

Kazman, R., & Bass, L., & Abowd, G., & Webb, M. (1993). Analysing the properties of User Interface Software. Software Engineering Institute, Carnegie Mellon University Technical Report CMU-CS-93-201. Pittsburg, PA.
Ouadou, K. (1994). Towards adaptive user interfaces. MIS-LISPI Laboratory. Doctoral thesis. Ecole Centrale de Lyon (France).
Pfaff, G. (1985). Seeheim workshop on User Interface Management System. Edited by Springer Verlag, New York.
Seffah, A. (1993). Proposals for an integrated courseware environment : Concepts, Methods and Tools. MIS-LISPI Laboratory. Doctoral thesis. Ecole Centrale de Lyon (France).,

A Case Study of Hypermedia Assisted Learning in "Life Environment Studies" II

Seiichi Shimizu
Hyogo University of Education, Katougun, Hyogo Prefecture, Japan

Seiichi Kisi
Kibi Elementary School, 256 Niwase, Okayama city, Japan

From April 1992, new curriculums have come into practice in Japanese elementary school and "Life Environment Studies" took effect at the same time. Most important object of "Life Environment Studies" is to support the growth of children's independence. To realize of this object teachers need to change their past typical teaching for all students in class. In "Life Environment Studies", students need to study with strong consciousness about the problem and to have their solution concerning of the problem. So the most necessary ability bringing up in "Life Environment Studies" is for children to investigate with their problems.

Meanwhile characteristics of hypermedia as Media are non-linear, non-continuance, non-structure and control by learner. Considering merits of hypermedia, it is very effective for children to study "Life Environment Studies" applying hypermedia.

In this paper ,we would discuss about hypermedia as one of the most effective materials to realize the object of "Life Environment Studies " and to lighten the burden imposed on teachers in the process of development of hypermedia materials.

Software development and practice in teaching

Computer Assisted Instruction(CAI) which was bound by text and teacher's guidance has strong historical roots in Japanese Education. But in the learning style of CAI, recently there is growing recognition of the need for changing to the children-focused learning environment . Hypermedia has been represented as new and powerful tools to help and support the transformation of Japanese classrooms. Children can study not by force but independently their own mental curiosity by applying hypermedia strategies. In "Life Environment Studies ", it's most important for children to interested in relationships between themselves and near society ,nature. So hypermedia materials is more useful than other materials to study for student's divergence of thinking.

We created stacks by HyperCard from the numerous computer files (text, graphics, audio, movies etc.). Each data of the card could linked other data of the appropriate cards in the stack. Clicking the "button" of the card by a mouse , children who use the stack easily enabled to dynamically search for their needful various type of information which is text, graphics, audio and movies. Movies are shown another TV monitor and they are easily controlled by HyperCard through the interface.

We used the development stacks in teaching 2nd grade class. After the lesson, we got to know that children were very excited and strongly motivated to study with the hypermedia System Most of all children had the impressions that the lesson was very interesting and they wanted to go the place to get more information. The main findings are the following:

F1 We can create stacks more easily and shortly than developing traditional CAI system.

F2: It make children possible to study along their individual interest because of the structural flexibility and interactive operation of the stacks.

F3 The stacks are designed to hold various type of data. Children can study more independently by operating these data as they like.

References

Ministry of Education (1989): National curriculums guidance for elementary school, edited by Life Environment Studies

C.Locatis. G.Lletourneau. & R.Banvard. (1986): hypermedia and Instruction, Educational Technology Research and Development, vol.37, No. 4, 65 77

Reaching Students through Bilingual, Interactive Multimedia Books

THOMAS J.C. SMYTH AND MARGARET R. CORBOY
School of Education
University of South Carolina-Aiken, 171 University Parkway, Aiken, South Carolina 29801

Interactive multimedia books, also known as CD-ROM storybooks, greatly appeal to educators who wish to stimulate the reading interests of beginning readers and reluctant readers. The sound effects, music, illustrations and speech associated with such electronic literature bring stories alive for the reader by presenting a dynamic retelling of well-known children's books. The multi-lingual options offer additional appeal to students who are learning a second language, for several of this genre of books present retellings in English and another language, most often Spanish.

Interactive multimedia books have been used successfully with elementary grade students who are native speakers of Spanish and with students who are learning Spanish as a second language. The student may elect to read and hear a Spanish or English version of the text. Likewise, he/she may select specific vocabulary from the text or from the illustrations to hear in both Spanish and English, thus allowing him/her to review pronunciation and to build vocabulary. For example, while reading Grandma and Me by Mercer Mayer (Brøderbund), the student may repeat any part of the text in order to hear Spanish or English pronunciations as well as definitions. In The Tale of Peter Rabbit by Beatrix Potter (Discis), the student may select text or significant parts of the illustrations to hear pronunciations in either language. The interactive nature of the reading allows the student to control the rate and the format of presentation and to repeat any part of the story as necessary.

These electronic books are ideal for students who have not yet discovered the joy of reading or who are learning a second language. Some criticism has accompanied the proliferation of such electronic storybooks, however. The genre is viewed by some as nothing more than excessive stimulation for the reader who engages in an MTV-like viewing of a story by passively listening and watching. These critics would suggest that the reader should be focusing simply on text and the imaginative images that it can induce. To the contrary, the dynamic interaction with text and images permits the reader to respond to the text by clicking on words, passages, or illustrations to make them come to life. Furthermore, the reader builds vocabulary and improves comprehension by monitoring his understanding of the passage easily and individually with no pressure from instructor or peers. In addition, several editions of these electronic books are accompanied by the books themselves so that the student may have the traditional reading experience as well, either before or after participating in the electronic version.

In order for bilingual, interactive multimedia books to be effective in developing students' understanding and appreciation of text, they must allow students to read independently and to explore stories interactively, controlling the pace of the program and choosing to read on their own. Furthermore, the best of this genre also encourage extension activities such as writing and oral discussion in response to the reading. Finally, for students who are learning a new language, many of these CD-ROM storybooks provide stimulating opportunities for independent language exploration which result in a non-threatening and thus natural acquisition of a new tongue.

"Physics by Pictures"
New Interactive Learning Environment for Physics Education

N. SOBOLEVA, S. KOZEL, P. ZHDANOVICH, K. GALICHSKY, M. ENTIN
Scientific Center PHYSICON
Moscow Institute of Physics and Technology
Box 13, Dolgoprudny-1, Moscow region, 141700, RUSSIA
E-mail: sob@scph.mipt.msk.su

Educational computer courseware "Physics by Pictures" is an integral base of knowledge in the field of physics which provides an interactive learning and teaching environment. "Physics by Pictures" includes a set of dynamic computer simulations of physical experiments and a reference book. The courseware includes various information and functions using simulation, multi-media, gaming, testing etc.,

"Physics by Pictures" is an educational computer courseware for both teachers and schoolchildren developed in Scientific Center PHYSICON (founded by Moscow Institute of Physics and Technology and Russian Physical Society).

"Physics by Pictures" consists of a reference books in physics and a set of computer programs. It includes colorful computer simulations from Mechanics, Thermodynamics and Molecular Physics, Electricity, Optics, Atomic Physics, as well as portraits and biographies of famous physicists, historical experiment examples, etc.

Some questions and various problems are included in the courseware with the possibility to enter and check the answers of schoolchildren. It is possible to choose all parameters of computer simulations. Some of the examples are designed as a flexible constructor, which allows one to develop different experimental schemes and to investigate them. There are also a calculator, lists of physics and mathematics formulas, and tables of physical constants. The hypertext is used to ease learning physics.

The "Physics by Pictures" courseware is intended for schoolchildren with different levels of physics knowledge. It can be used as a convenient reference book for the first level physics, and for deeper learning of physics at higher levels.

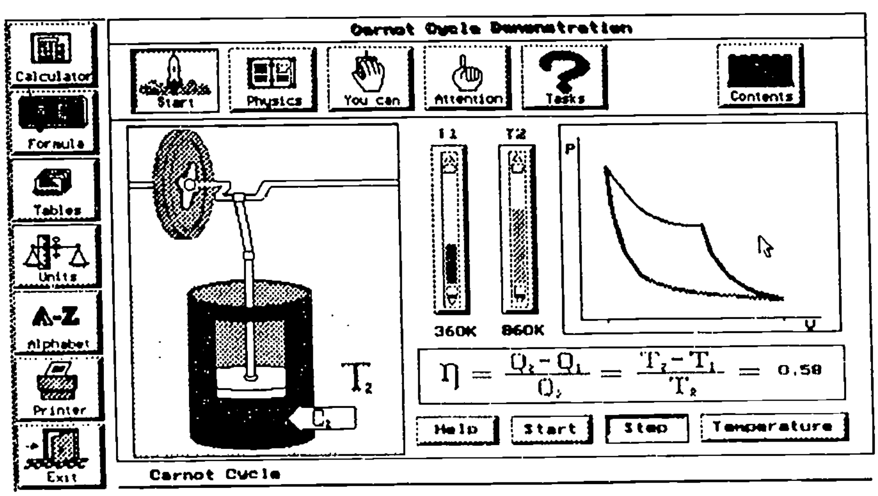
The distinguishable features of "Physics by Pictures" are:

a) the courseware includes various information and functions using simulation, multi-media, gaming, testing,

b) a friendly interactive graphics interface,

c) the courseware provides wide possibilities for more intensive teaching and attracts considerable interest of schoolchildren.

Hardware requirements for the courseware: IBM AT/286/386/486 compatibles, EGA or VGA graphic card, Microsoft compatible mouse, MS DOS 3.3 or higher. Now the new version of the software for Windows 3.1 is in progress.



Intelligent Performance Support for Courseware Authoring

J. MICHAEL SPECTOR
Armstrong Laboratory, Technical Training Research Division
7909 Lindbergh Drive
Brooks AFB, TX .78235-5352

Courseware authoring (developing computer-mediated instructional materials) consists of a variety of tasks often performed by several individuals. Some of these tasks are repetitive and involve skills that are easily and quickly acquired (e.g., invoking a particular computer-based authoring environment to open and edit a file containing a particular lesson module). Other tasks are less mechanical and require years of practice to achieve mastery (e.g., specifying a particular instructional strategy for a given instructional purpose and setting). Still other tasks require collaboration in order to insure effectiveness of solution (e.g., planning the use of interactive media so as to engage learners) (Rowland, 1992).

Instructional design intelligence consists of expertise in one of more of the tasks involved in the courseware authoring process. This suggests that there are multiple instructional design intelligences and, therefore, multiple ways to represent courseware authoring intelligence in a computer program. An intelligent courseware authoring performance support system could be categorized and evaluated by the type of intelligence provided. At least these three types are possible: (1) intelligent tool set (e.g., brings the right tool set to the job arranged in likely order of use); (2) intelligent design advisor (e.g., offers one or more solutions elaborated for similar instructional problems); and (3) intelligent design problem-solver (e.g., presents the user, perhaps a novice designer, an executable framework for solving the user's design task) (Duchastel, 1990; Spector, Polson, & Muraida, 1993).

Evaluating Intelligent Instructional Design Support Systems

It is likely that different evaluation concerns arise with various types of courseware authoring support systems. For example, for those systems which purport to provide an intelligent set of tools, then it would be reasonable to examine some task analytic hypotheses (e.g., providing dynamic lesson plans which adjust to type of lesson objective will enhance productivity) as well as various human-factors hypotheses (e.g., providing drop-and-drag reusable objects in a particular window enhances productivity). Likewise, in evaluating an intelligent courseware design advisor it might be appropriate to examine various cognitive hypotheses (e.g., providing case-based, context-sensitive elaborations of specific instructional design guidelines enhances understanding of courseware design processes, with effects on improved instructional design performance). Evaluating an intelligent design problem-solver would include evaluation of products with regard to instructional effectiveness. Evaluating a collaborative courseware authoring performance support tool presents new and unsolved challenges.

References

Duchastel, P. C. (1990). Cognitive designs for instructional design. *Instructional Science,* 19(6), 437-444.

Rowland, G. (1992). What do instructional designers actually ? An initial investigation of expert practice. *Performance Improvement Quarterly*, 5 (2), 65-86.

Spector, J. M., Polson, M. C., & Muraida, D. J. (Eds.) (1993). *Automating instructional design: Concepts and issues*. Englewood Cliffs, NJ: Educational Technology.

Multimedia Portfolios: Tool for Classroom Assessment

MARY L. STEPHEN
Department of Arts and Sciences
Harris-Stowe State College, 3026 Laclede Ave., St. Louis, MO 63103, USA

As the use of technology in educational settings increases, traditional methods of assessment such as written tests and reports often prove to be unsatisfactory. Newer technologies appeal to a variety of learning styles, while the familiar methods of assessment do not necessarily address these differences. With the availability of low cost, relatively easy to learn multimedia authoring systems, traditional methods of assessment can be replaced by more innovative means of assessment. Multimedia portfolios that contain samples of student work and document student progress offer one alternative to the traditional assessment tools. The use of such portfolios poses several challenges for the teacher. These include questions of evaluation, and selection of activities for inclusion that do not simply duplicate traditional written forms of assessment.

This project focuses on the use of multimedia portfolios as assessment tools with two different populations. The first group consisted of first grade students from an inner city school. These students created multimedia math journals to illustrate concepts studied in their class. Journals included original artwork, clip art, buttons, text items, digitized pictures and verbal comments recorded by the students. Cards selected from individual journals were also combined into a class math journal. One unit included counting in other languages, and students were able to include in the journal recordings of themselves counting in these languages. An advantage of this type journal is the ability to record a student's spoken explanation of a math concept. Much of the artwork and writing could have been produced with paper and pencil. The ease of duplicating clip art made the students more willing to work with larger numbers in their examples, and the ability to resize clip art using the program reinforced the concept of relative sizes.

The second population consisted of preservice and inservice teachers in a college technology course. These students created multimedia portfolios that illustrated uses of technology in education. Students created simple stacks explaining, and when feasible, demonstrating output from a variety of technologies such as scanners, digitizers, video discs, CD-ROM, and MIDI music. Text items describing the use of these technologies in educational settings were included in the stacks Creation of a portfolio in this manner enabled the students to learn how to use a hypermedia authoring system as a tool. Inclusion of samples of the technology in the stacks required the students to demonstrate proficiency with each of the types of technology. Students enjoyed this approach and felt it demonstrated what they had learned more effectively than traditional written assignments would have demonstrated. At the end of the semester, students used their stacks to create videotapes on technology in education. This provided an alternative for those students without access to a computer after the course ended to continue to use the material.

Both groups used HyperStudio to create the portfolios. Samples of portfolios produced by both the first graders and the preservice and inservice teachers will be displayed.

Software: *HyperStudio*, Roger Wagner Publishing, Inc., El Cajon, California.

Authoring and Algorithmic Navigation in Large Hyperbases

DMITRY SUBBOTIN
Informatics Center
Moscow State Institute of Electronics and Mathematics
3/12, B. Vuzovsky str., Moscow 109028, Russia
E-mail: subbotin@icsti.msk.su

The problem of creation of large educational hyperbases and navigating them is especially vital for loosely-structured subjects, e.g. the humanitarian ones. The possibility of representing such a subject from various points of view is very significant. It is often difficult to define the limits of a material, or its "beginning" and "end". For these fields the hyperbase should not be organized as plain electronic textbook. The main intention of it is to provide for a student the possibility to perceive the material as it is, in multiple aspects, and to promote for self-learning and free navigation.

When a hyperbase is intended for the free navigation, linking of nodes in it is a very complicated and ambiguous problem. It is hardly possible to perform linking manually, not only because of the large quantity of nodes, but also because it is difficult to choose the unified criteria of linking. You never know in advance what will be the place of a particular node in various paths while in navigation. The pragmatic approach would be to link everything close by sense; in other words, each pair of nodes which *could* be joined with some linking expression or just could follow one another in a navigation path *should* be linked. The authoring system should provide facilities for automation of linking, performing search for nodes-candidates. The methods of information refining and powerful linguistic engines can certainly help in this matter.

The probability of getting lost in this hyperbase is very high. Of course, the "trail blazers" can do a lot of work, paving "good" paths in the base, but the ability of free navigation is also necessary.

On this stage, the automatic (algorithmic) navigation methods are suggested. The algorithms use only the structure of links in the hyperbase. They are based on the correspondence between topologic and semantic connectedness of nodes in a hypertext net. Speaking about topologic connectedness, we mean that every node has an unique position in the network (number of links, distances from other nodes, etc.). When the links are established by content, the topology of the network corresponds to the semantic interrelations of the nodes.

On each step of navigation the possibility exists to choose the next node upon criteria of high degree of its topologic connectedness with all the nodes previously included in the navigation path. The resulting navigation path forms a logical exposition of the theme set by the initial nodes.

The authoring and algorithmic navigation means are demonstrated on the IntelText system (Subbotin & Subbotin, 1993), which was developed basing on the abovenamed principles. IntelText works on the PC under DOS or Windows.

In IntelText, a hypertext node is a separate text fragment expressing one thought or idea or one small section of a textbook - in other words, a monosemantic unit. The links are bi-directional and go from a node to a node as a whole. Linking can be performed manually or automatically or in semi-automatic mode.

The navigation has three modes: manual, when the user traverses among nodes and can remember the navigation "path"; automatic, when the user sets the theme and the system automatically picks up the nodes and arranges them in a linear text according to the built-in criteria; and the combined mode, when the result of the automatic navigation is decompiled into a path which can be explored manually.

Thus, the learner has the following capabilities: traverse from one node to another, ask the system to automatically compile a path beginning at any particular node, walk along this path, look at the neighborhood of the nodes in it, leave the path for independent, manual navigation, return to the path, etc.

The author has additional capabilities of establishing and changing the links structure, remembering good paths, etc. He/she also can ask the system to construct paths automatically, consider the result and accordingly correct some links, thus improving the base.

References

Subbotin, M., & Subbotin, D. (1993). IntelText: producing coherent linear texts while navigating in large non-hierarchical hypertexts. *Lecture Notes in Computer Science, N 753, Springer-Verlag*, 281-289.

AlgoBlock: A tangible programming language for collaborative learning

Hideyuki SUZUKI and Hiroshi KATO
Information Technology Research Laboratories, NEC Corporation
1-1, Miyazaki 4-chome, Miyamae-ku, Kawasaki, Kanagawa 216, JAPAN

AlgoBlock as an conversational tool

In the belief that learning through discussion and cooperation is more effective than that in isolated environments, the authors intend to design a computer-aided learning environment in which primary and secondary students can improve their skills for planning and logical thinking through social interactions. The authors have designed AlgoBlock system (Suzuki et al. 1993) as a conversational tool that facilitates discussions among learners and activates collaborative programming. It is a set of tangible blocks that can be connected to each other manually to form a program. Each block corresponds to a command of a programming language. Learners can control a graphically displayed submarine on the CRT through Logo-like commands.

Collaboration on the tangible tool

With this tangible tool, learners can manipulate the programs manually and directly, and can share them on their collaborative work space. We believe that tangibility of AlgoBlock helps learners build programs through social interactions. To investigate the effect of the tool on learners' collaboration, we have had an observational session, in which three 12-year-old students were engaged in group programming works using AlgoBlock for 90 minutes. Results of the observation eloquently tell us that the tool is qualified as a communication facilitator. Following is a typical conversation in the session.

Subject1[01] " This " {Tried to insert Rotate-block between GoForward-block and GoRight-block [action-1]}
Subject2[02] " No! We only have to change this (parameter) to 3 " {Stretched out his hand toward blocks [action-2], and removed Rotate-block that Subject1 connected [action-3]} "Be...Because this is....."

(1)**Easy operation:** Handling tangible blocks is so direct and easy for Subjects that they could manipulate programs immediately when they wanted to [action-1][action-3]. Easy operation promotes trial-and-error activities that encourage conversations among learners.

(2)**Social display of ideas:** In [01], Subject1's idea was socially displayed to the others by the obvious action [action-1] and physical changes in configuration of the tangible blocks that were placed on shared work space. Subject2 thus could understand the idea easily and intervene immediately [02]. Subject2's response here is also displayed to the others [action-3]. In this way, manipulating the tangible tool necessarily generates sequence of social display of ideas. AlgoBlock works as an open tool (Hutchins 1990) which enables users to monitor their cooperative activities mutually. This feature contributes to support collaboration.

(3)**Gesture based turn-taking control:** AlgoBlock also enables gesture based turn-taking control to access the tool because of its tangibility. In [02], Subject2 stretched out his hand toward the blocks [action-2] to express his idea. By the external gesture that was naturally generated in trying to manipulate the tangible tool, the turn-shifting was smoothly achieved and confirmed. The gesture based turn-taking control supports vigorous collaboration because the rules, which are the same as that in everyday activities, are so familiar to learners that they do not have to learn extra rules for the control.

References

Hutchins, E. (1990) The Technology of Team Navigation. *Intellectual Teamwork*, Galegher, Kraut and Egido (Eds.), Erlbaum, 191-220.

Suzuki, H. & Kato, H. (1993) AlgoBlock: A tangible Programming Language -A Tool for Collaborative Learning-. *Proceedings of the 4th European Logo Conference*, 297-303.

780

Multimedia and Teacher Training: A Videodisc Application For Exploring Conceptions of Effective Teaching Practices

KAREN SWAN, CARLA MESKILL, and DAVID CHAPMAN
State University of New York at Albany, Albany, NY 12222

MARCO MITRANI
Interactive Generation, Box 183, Marlboro, VT 05344

Teachers' (and students') conceptions of effective teaching practices play an important role in the teaching and learning process, but are typically given footnote status in teacher preparation programs. This lack of attention to conceptions of effective teaching may have particularly critical repercussions, as when teachers are trained and/or curricula designed with a particular sort of teacher-student interaction in mind, then injected into cultures in which teachers and students are accustomed to interacting in dramatically different ways. The purpose of an on-going project at the University at Albany is to develop materials that explore differences in perceptions of effective teaching practices and that stimulate preservice teachers to reflect on their preconceived notions of the same. Primary among these materials is a videodisc which presents loosely scripted interactions between a teacher and her students.

Teachers' perceptions of effective teaching are subjective. They are influenced not only by beliefs about the effectiveness of particular teaching strategies and styles (factors we are interested in exploring), but by expectations concerning classroom environments and what should be covered in particular disciplines, as well as by affective reactions to the personalities of differing teachers and students. These latter biases may be particularly troubling in cross-national studies. Although video is generally recognized as a useful tool for exploring teaching effectiveness, most of the video used for such purposes consists of tape made in actual classrooms, making it difficult to partial out differences in perceptions based on differences in teaching strategies and styles from differences in perceptions arising from other factors. Our approach, therefore, has been slightly different.

The video used in the work discussed here was designed to focus on particular teaching strategies and styles while holding other factors constant. It consists, therefore, of quasi-scripted vignettes of a teacher and her students exploring a non-traditional subject area, black history. The teacher and all her students are African-American. The segments were shot in a circular meeting area in a local community school. Two cameras were used to capture both teacher and student behaviors, edited into segments which vary by teaching strategy and style, then put on videodisc. The use of videodisc allows for the random presentation of these segments.

We are currently using the videodisc in two ways. Firstly, it is being used in a cross-national study of conceptions of effective teaching practices. We are showing the vignettes to educators from a variety of different countries and asking them to rate the effectiveness of the teaching shown in each on a ten-point scale. We are using comparisons of response patterns, in particular comparisons of the responses of educators in the United States with the responses of educators in African countries where we are working, to inform our international curricular work. In addition, we are comparing the responses of preservice and inservice teachers and teachers and students in this country. Because the vignettes are on videodisc, we can also explore the effect of presentation on responses.

Secondly, we are using the videodisc in our teaching methods classes as a prompt for discussions of teaching strategies, styles, and reflective practice. We are finding that the vignettes elicit a rich discourse around these topics, topics that have not traditionally been as well covered as we believe they should be in preservice training. The use of videodisc in these classes allows us to explore particular parts of each vignette in microcosm and to relate different vignettes to each other without struggling with videotape. Students' response to these discussions have been very positive, so much so that we are currently considering integrating our more conventional videotape examples into a similar structure for teaching purposes.

Interactive Multimedia for Preservice Elementary Teacher Education

T. E. Thompson
Department of Curriculum & Instruction
Northern Illinois University, DeKalb, Illinois, 60115, USA

Six videodiscs were developed for use in a preservice elementary education science methods course. Each contains a twenty minute hands-on science lesson taught by an exemplary teacher, and a reservoir of thirty second episodes that also deal with the specific topic of the videodisc. The disc titles include: Process Skills I, Process Skills II, Interdisciplinary Science, Methods of Instruction, Classroom Management, and Questioning. Each disc contains four separate sound tracks: 1) the lesson as it happened, 2) the teacher's comments, 3) the childrens' comments, and 4) a science educator's comments.

This project was designed to facilitate instruction in science education. Current trends in education call for providing meaningful instruction. In science education this means hands-on activities that allow students to construct their own meanings. This series of videodiscs is an opportunity to bridge research into practice. It provides an opportunity to capture positive role modes implementing outstanding pedagogy in their practice.

Six elementary school teachers (grades 2-7) were identified as exemplary science teachers. This process began in the fall of 1992. After the teachers were recruited they met at Northern Illinois University where they were inserviced with regard to the project. Philosophy of the project, science methodology, process skill acquisition, questioning strategies, and videotaping details were discussed.

After the initial taping a rough edited version of the 1/2" betacam videotape was created. Upon completion the videotape was taken back to the school for the children to view. Their comments regarding what they remembered taking place were recorded.

The teachers were brought into the production studio at the university to record their comments. The science educator's comments were also recorded at the production studio. At this point two 3/4" parallel masters were produced for each lesson. One tape contained the video and the sound tracks for audio one and two. The second tape with identical timecodes contained the sound tracks for audio three and four.

The discs are currently being used in Level I applications while the software is being developed. Elementary science methods instructors are integrating selected video segments into lecture/discussion sessions. The two objectives are to provide role models for preservice teachers and to bridge the gap between theory and practice. Access is obtained through the use of a remote keypad.

Preservice teachers are provided with sets of barcodes that correlate with class discussions and textbook readings. Access is obtained through the use of a barcode reader. The videodiscs are located in the Science Education Laboratory and are available to students outside regular class time.

LinkWay Live is the software selected to develop the interactive multimedia component. When completed, preservice and inservice teachers alike will be able to interact with all six videodiscs in the following ways: 1) A lesson plan containing standard elements such as concept and process objectives, procedures, key questions, etc. serve as the primary shell for each featured science lesson. The viewer will be able to click on any icon found within the lesson plan and view the application of the plan and the resulting behaviors. 2) Assessment issues are another focus of the software. Opportunities to correlate with the Illinois State Goals for Learning, the American Association for the Advancement of Science document, Benchmarks, and common evaluation techniques will be provided. 3) Decision making opportunities allow the viewer to make choices about instructional alternatives that he/she might make during an activity.

Two additional videodiscs are currently in production on the topics of authentic assessment and technology. By the end of 1994, all eight videodiscs under the title: Capturing Excellence will be placed in the hands of the science educators involved in teacher preparation programs in the state of Illinois. Beginning in September, 1994, the project director will visit each Illinois public institution involved in teacher education to conduct staff development activities and disseminate the videodiscs.

Domain Analysis for CAI Design

LORNA UDEN
School of Computing
Staffordshire University, The Octagon, Beaconside, Stafford ST18 0AD, England.

Educational software currently being produced suffers from severe design flaws which result in poor quality. For designers of Computer-Assisted Instruction to produce effective instruction, it is necessary to provide a set of procedures and guidelines that designers can follow, based on sound instructional design and learning principles.

A methodology known as GOMS can be used to assist designers during the first phase of the design model [Roblyer 88]. Determining what is learned is the Goal of the design. Lessons are designed to accomplish goals for specific learners. Having identified the instructional goal, it is necessary to determine the type of learning **O**utcomes required of the student. According to Gagne [92], learning outcomes can be placed into five categories. Each requires a different instructional method to accomplish it effectively. Since the underlying structure of the instructional content differs by domain, determining the domain of learning outcomes gives the CAI designer an initial idea about organising and sequencing instruction contents for teaching those outcomes.

Different Methods must be used to analyse subject matter in different domains because of different learning outcomes. There are four different methods that can be used for analysing different learning outcomes.

For educational software to be effective, what is needed is the incorporation of higher order learning Strategies within the courseware. Learning strategies include techniques that learners can be taught to use during instruction to support effective learning [Uden 93]. Good guidelines and design methodology can enable designers to achieve effective courseware production. The GOMS methodology offers such an approach.

References

Dick, W. & Carey, L. (1991). The systematic design of instruction. Glenview, Ill: Scott-Freeman.

Roblyer, M.D. (1988). Fundamental problems and principles of designing effective courseware. In Instructional design for microcomputer courseware. (Jonasson, D. ed.) LEA.

Uden, L. (1993). Improving learning ability in computer-assisted education. In 1993 Instructional conference on computers in Education, Dec. 15-17 1993. Taiwan, R.o.C.

"Power Users" in Academe: Who are They?

T. L. VAN VALEY
Kercher Center for Social Research
H. POOLE
Division of Media Services
Western Michigan University, Kalamazoo, Michigan, 49008, USA

M. VAKALIS
Department of Mathematics and Computer Science
Capital University, Columbus, Ohio, 43209, USA

With respect to computing, the people who might be labelled "Power Users" are an important group in any organization. They often provide the necessary internal training (formal and informal) that an organization needs in order to respond adequately to rapid technological change. Moreover, they are often models for computing resource development within their organizations. In addition, by learning how to respond to the challenges of the new technology, they are typically pioneers who smooth the way for those that follow. Finally, they often occupy leadership roles in the decision-making processes regarding the integration of computing within their organizations. This paper suggests a set of criteria that can be used for the definition of "Power Users" in academic environments. It also describes the "Power Users" that were identified in a survey of the faculty at Western Michigan University, and discusses some of their attitudes about and perceptions of academic computing.

Toward a Definition of Academic Power Users

Based upon the literature, a number of criteria were identified that are central to the notion of "power users." These included: technical expertise, amount of time spent using a computer, the range of applications used, the expertise of use, and integration into a larger group of skilled computer users. Based on a 1989 survey of 500 faculty at Western Michigan University, six criteria were tentatively established as the initial definition of "power users:" 1) Total time working on a computer in excess of 15 hours per week; 2) Use of a word processor a few times a week or more; 3) Use of electronic mail; 4) Use of at least one application in addition to word processing; 5) Teaching or expecting to teach classes that require the use of computers by students; and 6) Purchase or acquisition of a computer for use at home or at the office.

To assess the relative importance of each of these criteria, we simply removed each of the six criteria in reverse order and observed the consequences. For example, when the sixth criterion "using one other application" was removed, exactly the same number of "power users" was produced (n=54). Since it makes no difference, clearly this factor is redundant to the definition. Also, since the fifth criterion "bought/acquired a pc" affects only one person, it is also redundant. Indeed, even the fourth criterion, "using a wordprocessor" affects only 4 individuals. In contrast, removing the third criterion "taught classes" increased the number of tentative "power users" to 72 (an increase of nearly 25 percent). Clearly, this criterion should be retained. Since the other two criteria accounted for the most Power Users, the final definition of "academic power users" among the faculty at Western Michigan University in 1989 should be based upon the first three criteria in the definition:

1) the use of electronic mail;
2) at least 15 hours of computer use per week;
3) teaching or expecting to teach classes that involve the use of a computer.

Although Power Users currently make up a small percentage of computer users (perhaps 10%), they are the prototypical faculty computer users of the future. What only the power users are doing with computers today will become commonplace among faculty in general within a few years.

Techniques for Converting Analog Interactive Videodisc Courseware into Digital CD-ROM Programs: Conversion of *The Anatomy Project*

Alan VanBiervliet, PhD
Professor & Dir. of Research
Dept of Physical Medicine & Rehab
Univ. of Arkansas for Medical Sciences
Little Rock, Arkansas 72205
Phone: (501) 661-1202 ext. 2373
VANBIERVLIET@LIFE.UAMS.Edu

Thomas R. Gest, PhD
Associate Professor
Office of Ed. Dev & Dept of Anatomy
Univ. of Arkansas for Medical Sciences
Little Rock, Arkansas 72205
Phone: (501) 686-7054
TRGEST@Life.UAMS.Edu

During the past decade, educators in the health sciences have effectively and widely used interactive videodisc technology for the delivery of instructional material. Many videodiscs have been produced that contain very high quality still and motion images of anatomy, physiology, and medical procedures. *The Anatomy Project* is a joint British and American effort to produce a series of high quality instructional programs that uses multimedia technologies to combine technical content in an artful format. Currently, a set of seven videodiscs and programs exist that are used in medical schools around the world. When completed, *The Anatomy Project* will contain 26 volumes of interactive videodisc courseware, covering all of clinically oriented human anatomy. Recent advances in CD-ROM and digital video technologies enable high quality digital adaptations of these interactive videodisc programs to be created and presented using personal computers.

A large number of critical decisions must be made before attempting to convert an interactive videodisc program into a digital format. One of the most important is the hardware requirements for using the program. Many medical schools have already invested heavily in a variety of computer equipment for instructional purposes. One goal during the conversion of *The Anatomy Project* was to redesign the program to take advantage as much as possible of existing installed bases of equipment. These decisions included designing for 14 inch monitors, requiring only 5 MB of RAM, and requiring no additional display or conversion hardware (i.e., software only decompression). In addition, it was decided to design for double speed CD-ROM drives since they have replaced single speed drives as the computer peripheral standard. The tools used in the conversion from analog videodisc to digital CD-ROM included a video capture and compression board (VideoVision Studio™), Adobe Premiere™ and VideoFusion™ video editing software, HyperCard™, QuickTime™ compression and playback, Adobe Photoshop™, and various graphics programs.

Obtaining good performance from digital programs stored on CD-ROM (i.e., smooth video playback with synchronized audio) requires the developer to carefully consider and manipulate a wide range of complex system and media factors. Some of these factors are: CD-ROM bandwidth; video frame rate and resolution processing; video decompression processing; video display processing; audio sampling rate and bit depth; audio playback processing, processing time for "de-interleaving" and scheduling video and sound; and flattening. A number of techniques have been developed to enhance movie playback. One of the most useful involves capturing a video frame as a still image, digitally enhancing the image, and then inserting copies of the still image in place of the original video segment. A second technique involves replacing full motion video with a montage of still images while maintaining the sound or narration.

Recombinant DNA Lab
An Interactive HyperCard Simulation

RODRIGO VIVANCO B.C.Sc.
InfoMagnetics Technologies Corporation
11-1329 Niakwa Road East
Winnipeg, Manitoba, R2J 3T4, Canada
email: rodrigo@infomag.mb.ca

FRANCISCO MADRID Ph.D., B.Ed.
AgroBiotech Inc.
35-Mount Allison Bay
Winnipeg, Manitoba, R3T 3L4, Canada

Within the last decade biotechnology, the techniques and procedures used in genetic engineering, has become increasingly important and well known in universities and industry. However, this knowledge has not crossed over to the general public or even high school students moving towards a scientific career. The HyperCard stack described in this paper takes the theoretical concepts and experimental procedures of biotechnology and incorporates them into an interactive computer simulation of a basic biotechnology application. The student is given the task of producing human insulin using genetic engineering techniques.

The simulation takes place in a virtual laboratory where the student can move from room to room and work station to work station at the click of a button. The student starts the experiment by preparing the vector DNA to be spliced with the human gene. The student selects the bacterial culture which has the desired plasmid and uses enzymes to release the plasmids. From there the student isolates the plasmids using a centrifuge and then selects the appropriate restriction enzyme to cut the plasmid DNA at the desired DNA sequence.

The next task for the student is to prepare the insert DNA by first extracting mRNA from human pancreas and using reverse transcriptase to create the DNA sequence which produces insulin. The insulin gene is prepared for splicing by attaching the regulatory sequence to the gene. This will activate the gene and allow the host cell to produce human insulin. The final procedure after the insulin gene and plasmid are spliced is to introduce the hybrid vector into bacterial cells. The student then screens for bacterial cells which have accepted the hybrid vectors and will naturally produce insulin. These bacteria are then placed in a culture broth where human insulin is harvested.

Though the necessary steps are outlined in the protocol stack, the student interactively performs the tasks using equipment that would be found in a real laboratory. In the protocol stack there are links to supplementary stacks which further explain basic concepts to the student. For example, there is a linked stack which explores the use of vectors in genetic engineering. Both viral vectors and plasmid vectors are demonstrated.

At all the work stations the student can click on a virtual microscope which shows an animation sequence to help the student visualize the biological processes that are happening inside the test tubes. The graphical nature of the laboratory along with the animations of what is happening at the cellular and molecular level provide an insightful and entertaining introduction to genetic engineering. This virtual laboratory is an affordable alternative to having a fully stocked biotechnology lab at every high school.

Acknowledgments

Science and Culture Canada for funding the project.
Dr. Richard Bochonko and the Astronomy Group, University of Manitoba.
Apple Research Partnership Program

Partners in Education and the Distributed Interactive Simulation Instructional Animation Project

MICHAEL D. WILLIAMS
Naval Air Warfare Center Training Systems Division
12350 Research Parkway, Orlando, FL 32826-3224
williamsm@ntsc.navy.mil

MARSHA VANDIVORT, JASON AHMANSON
Instructor, Student
Edgewater High School, 3100 Edgewater Dr., Orlando, FL 32804

The overwhelming ills faced by education today will never be cured by using outdated traditional processes for education. Worn-out lectures, tests and homework fall far short in challenging high school students to learn the skills they desperately need to face their rapidly changing future. The process is changing in Orange County, Florida. The Naval Air Warfare Center Training Systems Division (TSD) and Edgewater High School, supported by Apple Computer, Inc., have joined under the *Partners in Education Agreement* to provide a new learning paradigm in one classroom environment.

The Training Systems Division needed a method to explain the underlying concepts of Distributed Interactive Simulation (DIS) and Edgewater High School was looking for ways to utilize their computer animation lab. Edgewater and TSD jointly planned a learning venture for the students to produce the Distributed Interactive Simulation Instructional Animation. This project provided a "real-world" multi-media production that would enrich students' skills in visual arts, group dynamics, computer operation, and problem solving in a multi-disciplined team environment.

The primary objective was to make the entire project a creative product developed by students. The teacher and facilitating partners encouraged students to learn to structure a task from conception to completion, work in groups and independently, communicate ideas verbally and visually, manage time and set priorities. To provide the students with a non-traditional, highly effective learning environment, the organization of this student team was modeled after a real corporation complete with a customer, TSD.

The facts that the corporation had the capability to interview, hire, promote, or dismiss employees based on production needs and work performance, helped motivate all students to carry their share of the workload. The student and teacher held regular corporate meetings to make work assignments, prioritize time, delegate project responsibility, critique the animation, and evaluate progress. Student communication and organizational skills improved rapidly as they assumed their full corporate responsibilities.

With this type of partnership project, the teacher becomes a resource person, facilitator, and stage manager, creating a cooperative learning environment. The students develop effective working skills as traditional classroom assignments are minimized and more of the students' effort is directed by self-initiated tasks. The teacher serves as a resource manager, identifying and securing community people with specialized skills to enhance the learning production process.

Throughout the project's lifespan, subject matter experts assisted the students in the areas of engineering and DIS, computer technology, creative writing, theater, and visual arts. Parents volunteered clerical skills, transportation, food, technical skills, equipment, and family time. Their support was invaluable.

Students reported learning important skills from participation in this project such as: cooperation, drawing, color theory, organization, public speaking, advertising, brainstorming, animation, working with others, problem solving and business planning. This partnership effort allowed students to develop real world competencies such as imagination, flexibility, leadership, and responsibility. What started as a simple classroom project, evolved in into a revolutionary teaching and learning experience.

Application of the Russian Language Processor Russicon in Language Learning

S.A.YABLONSKY
Department of Computing St.Petersburg's Transport University, St.-Petersburg, 190031, Russia

New electronic media now provides fast and easy access to the authoritative language information. Natural language processing as a part of Artificial Intelligence is seen by some as one of the most promising directions in computer based language learning. In Intelligent Language Tutoring Systems of various types, natural language processing is one of the most important components. There are many popular linguistic applications for Romance and Germanic languages that could be used for language learning. Unfortunately, we can't confirm the same about Russian language processing, in which good results were achieved for IBM 360 (Popov, 1986) and only first steps are made for personal computers.

We shall describe the design of, and the possible applications of Russian language processor RUSSICON capable of acting as an intelligent system for analyzing and synthesizing of Russian text.

Processor RUSSICON (Yablonsky, S.A., 1990; Belyaev, B.M., & Surcis, A.S.,& Yablonsky, S.A. 1993) includes six main blocks: 1) Set of machine dictionaries. Base machine Russian dictionary consists of 36611 word-building stems. One word-building stem generates from 1 up 80 word changing stems. Wordbuilding stems of the dictionary give birth to 84167 word-changing stems. Word-changing stems in there term give birth to 3600000 different words. 2) System for construction and support of computer dictionaries. 3) Morphological analyzer allows to define following grammatical characteristics of a word: part of speech, changeability, animation, case, number, gender, person, aspect, tense, transition, mood, form, reflexive (verb), length of word-building and word-changing stem. 4) Normalizer which modifies a given word to its normal grammatical form (lemma). 5) Syntactic analyzer. 6) Semantic analyzer. The processor is designed as C library (Borland C++) of mentioned functions.

Several applications are build on the base of the processor. Russicon Russian Spelling Corrector checks wrong spelling. Russicon Russian Electronic Thesaurus provides users with complete synonym information for words or phrases. Russicon Russian Language Expert System provides all functions of processor, allows to receive morphological information of the word and to build normal form for the word, shows paradigm for the word, constructs new words lexicon, constructs frequency lexicon, provides morphological information treatment of new words not in the base dictionary. Russicon's Russian Electronic Dictionary for any form of input Russian word outputs a) one or several lemmas (lexical homonims); b) one or several sets (the case of morphological homonyms) of such grammatical characteristics: part of speech, case, gender, number, tense, person, degree of comparision, voice, aspect, mood, form, type, transitiveness, reflexive, animation; c) the synonym row(s); d) the antonyms; e) the precise definitions; f) the explanatory comments; g) the examples of usage etc. All linguistic applications could be used as a seperate systems or inside of some Tutoring Systems.

References

Popov, Eduard V. (1986). *Talking with Computers in Natural Language.* Springer-Verlag, 305p.

Yablonsky,S.A. (1990). Design of Russian Language Processor for Personal Computers. In *Proceedings of the II Union Conference Artificial Intelligence.* Minsk, SU.

Belyaev,B.M.,& Surcis,A.S.,& Yablonsky,S.A. (1993). Russian Language Processor RUSSICON: Design and Applications. *In Proceedings of the East-West Artificial Intelligence Conferense (EWAIC-93),* Moscow

Author Index

789

❖

AACE

ASSOCIATION FOR THE ADVANCEMENT OF COMPUTING IN EDUCATION

ISBN 1-880094-10-X