DOCUMENT RESUME

ED 360 949 IR 016 177

AUTHOR	Maurer, Hermann, Ed.
TITLE	Educational Multimedia and Hypermedia Annual, 1993. Proceedings of ED-MEDIA 93--World Conference on Educational Media and Hypermedia (Orlando, Florida, June 23-26, 1993).
INSTITUTION	Association for the Advancement of Computing in Education, Charlottesville, VA.
REPORT NO	ISBN-1-880094-06-1
PUB DATE	Jun 93
NOTE	673p.
AVAILABLE FROM	Association for the Advancement of Computing in Education, P.O. Box 2966, Charlottesville, VA 22902 ($45).
PUB TYPE	Collected Works - Conference Proceedings (021)
EDRS PRICE	MF04/PC27 Plus Postage.
DESCRIPTORS	*Distance Education; *Educational Media; Educational Technology; Elementary Secondary Education; Foreign Countries; Futures (of Society); Higher Education; *Hypermedia; *Multimedia Instruction; *Telecommunications

ABSTRACT

This collection presents papers pertaining to the wide area of educational multimedia and hypermedia. The ED-MEDIA conference series aims at using new technology for purposes beyond "infotainment" and "edutainment" to improve education for all. The volume is a guide to what is happening in educational multimedia and hypermedia right now and what is to happen in the future. Developments in elementary and secondary education, higher education, and distance education are reviewed. These proceedings contain 75 full papers, reports of 4 panel discussions, 19 short papers, and reports of 32 demonstrations and posters. (SLD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Educational Multimedia and Hypermedia Annual, 1993

ED 360 949

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it
☐ Minor changes have been made to improve reproduction quality
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
G.H. Marks
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR016122

AACE

BEST COPY AVAILABLE

Educational Multimedia and Hypermedia Annual, 1993

edited by Hermann Maurer

Proceedings of *ED-MEDIA 93*—
World Conference on
Educational Multimedia and Hypermedia
Orlando, Florida, USA, June 23-26, 1993

AACE Association for the Advancement of Computing in Education

ERIC

ISBN 1-880094-06-1

Copyright © 1993 by the Association for the Advancement of Computing in Education

A'' rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without permission in writing from the publisher.

The publisher is not responsibile for the use which might be made of the information containing in this book.

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

Printed in the USA

ERIC
Full Text Provided by ERIC

Preface

As chair of the Steering Committee of the ED-MEDIA conference series, I have the pleasure of presenting this timely collection of information pertinent to the wide area of educational multi- and hypermedia. I am doing so on behalf of the steering committee which also includes David Jonassen, *University of Denver, CO (USA)*; Gary Marks, *AACE, Charlottesville, VA (USA)*; Douglas Norrie, *University of Calgary, Calgary, Alberta (Canada)*; and Thomas Ottmann, *University of Freiburg, Freiburg (Germany)*.

As many of you are aware, the ED-MEDIA conference series resulted from merging the International Conference on Computer Assisted Learning (ICCAL) conference series with efforts of the Association for Advancement of Computing in Education (AACE) to offer a broad-based conference series with a strong emphasis on both educational and multimedia aspects. Those of us who have been involved in this merging process are exceedingly happy with the outcome: the timing could not have been better. The breakthrough of digital media is visible everywhere: the computer as the ultimate media-machine, combining textual and graphical data with interactive video and audio clips, is now a reality. Potent commercial forces are pushing this combination into the market to an extent that will reshape how we live, work, and learn—be it in the form of electronic books, encyclopedias, dictionaries, roadmaps or picture collections; be it in the form of "interactive movies" with the long awaited CD-I technology; be it in worldwide hypermedia networks, in small portable computers with CD-ROM or CD-I integrated in them, or in ever more powerful virtual reality settings.

This is an important juncture in the history of computers. Quite heady, intriguing, with much research and commercial potential, yet also with the danger that developments we are witnessing are not going to change our society for the better as we would hope, but might well add another ingredient helping to "amuse ourselves to death," quoting Neil Postman's famous book. Let us not forget that television was introduced with fanfare some 40 years ago as the ultimate tool to deliver education to everyone and has turned out to be (albeit with exceptions) a mind-numbing machinery, pacifying humankind and replacing education by superficial trivia information. Computer-based multimedia has, much more so than televison, the opportunity to indeed offer better training and education in many ways; however, the same technology may also lead to still more entertainment, to seducing people to leave everyday problems behind by submerging them into new technological worlds, into virtual realities.

ED-MEDIA is recognizing this challenge. We believe that a concentrated effort has to be undertaken to counteract pressures of the entertainment industry to mainly use multimedia for "infotainment" and "edutainment." The ED-MEDIA conference series aims at using new technology for more noble purposes: for the education of woman- and mankind. In doing so, ED-MEDIA will have to closely follow developments in many areas, from the electronic entertainment industry to artificial intelligence applications and worldwide networking. Yet, our concern is that all these technologies are used, to a large degree, for educational rather than other purposes.

Looking at this volume and at the ED-MEDIA 93 program gives us some basis for optimism. We have been able to assemble an impressive array of educational applications on the forefront of research and development.

It is, of course, almost an absurdity to present the newest developments in educational multi- and hypermedia in the form of a printed book. We are well aware of this and are contemplating alternatives or supplements for future ED-MEDIA conferences. Yet the lack of a unified platform makes this, as we all know, a formidable task! Thus we are proud that even without the newest technologies this volume manages to be an invaluable guide to what is happening in educational multi- and hypermedia

ERIC
Full Text Provided by ERIC

right now, or is about to happen in the near future. That this is the case is due to the cooperation of all authors, and to the tireless efforts of the Program Committee and many referees who tried to package multimedia know-how in printed form. They have succeeded. For this we would like to thank all involved, and are happy to acknowledge the work of the Program Committee by listing it below.

May this volume increase your knowledge in your particular area of interest! However, remember that the best way to learn is through various forms of participation as experienced through ED-MEDIA conferences, with their demonstrations, panels, workshops, and papers. Hence, see you (also) at Vancouver, Canada, for the next ED-MEDIA conference in June 94!

Hermann Maurer
Chair of ED-MEDIA Steering Committee
University of Auckland
Auckland, New Zealand (on leave from Graz University of Technology, Austria)

ED-MEDIA 1993 Program Committee

Philip Barker, *Teesside Polytechnic, Cleveland (UK)*
Ian D. Benest, *University of York (UK)*
Panels Chair: Marc H. Brown, *DEC Systems Research Center, Palo Alto, CA (USA)*
Patricia A. Carlson, *Rose Hulman Institute of Technology, Terre Haute, IN (USA)*
Neal S. Coulter, *Florida Atlantic University, Boca Raton, FL (USA)*
Gordon Davies, *Open University, Milton Keynes (UK)*
Thomas W. Greaves, *IBM, Atlanta, GA (USA)*
Kristina Hooper-Woolsey, *Apple Multimedia Lab, San Francisco, CA (USA)*
Wolfgang Irler, *University of Trento (Italy)*
David Jonassen, *University of Colorado, Denver, CO (USA)*
Demos/Posters Chair: Greg Kearsley, *George Washington University, Washington, DC (USA)*
Gyoesoe Kovacs, *SZAMALK, Budapest (Hungary)*
Gary Marchionini, *University of Maryland, College Park, MD (USA)*
Gary Marks, *AACE, Charlottesville, VA (USA)*
Chair: Hermann Maurer, *Graz University of Technology, Graz (Austria)*
Cliff McKnight, *HUSAT Research Institute, Edinburgh (UK)*
Max Muehlhaeuser, *University of Kaiserslautern (Germany)*
Douglas Norrie, *University of Calgary, Calgary, Alberta (Canada)*
Graham Oberem, *University of Washington, Seattle, WA (USA)*
Tim O'Shea, *Inst. of " ' Tech., Open Univ., Milton Keynes (UK)*
Setsuko Otsuki, *Kyushu Institute of Technology (Japan)*
Thomas Ottmann, *University of Freiburg (Germany)*
Workshop Chair: Thomas Reeves, *University of Georgia, Athens, GA (USA)*
Ivan Tomek, *Acadia University, Wolfville, Nova Scotia (Canada)*
Beverly Woolf, *University of Massachusetts, Amherst, MA (USA)*
Nicole Yankelovich, *Sun Microsystems, Billerica, MA (USA)*

❖

Table of Contents

Full Papers

ERIC

ERIC

PANELS

Short Papers

Demonstrations/Posters

❖

Full Papers

Multimedia: We have the Technology but do we have a Methodology ?

JAMES L. ALTY
LUTCHI Research Centre, Dept of Computer Studies, University of Loughborough, UK

Multi-media presentation systems are now becoming commonplace. They can be utilized on personal computers such as the MacIntosh or IBM PC, and are also available on workstations such as the SUN. Although there are some technical problems associated with synchronisation, with the handling of large moving pictures and with real time performance, most of these problems have been, or shortly will be, overcome. There is therefore no technological reason why large numbers of multimedia systems should not be successfully implemented during the next few years. The real difficulties, however, in developing multi-media presentation systems do not lie with the technology. They stem from a lack of a methodology for multi-media design. The real question is not HOW can we provide different media but WHEN and to WHAT effect (Alty 1991). In a sense, we have been in this position before. We constructed textual interfaces before we understood command languages. We developed graphical approaches without really understanding the essentials of graphics design and we used colour with a combination of arrogance and ignorance which would have embarrassed a true artist. There was some merit in using such a pragmatic approach in the past. Computing applications required new interfaces whose design was not fully understood or supported by research. It therefore it seemed reasonable to try out as many options as possible and select the best for further development. The problem with using the same approach for multimedia interface design is the size of the search space. We have so many options and combinations that the possibility of producing poor interfaces is high. It is therefore important to try to lay down design guidelines and principles to aid the multimedia application developer.

There are difficulties in simply extrapolating from previous work in single media. Marmollin (1992) has argued, for example, that the multi-media approach should be seen as a new approach to systems design rather than a new technology, and that our definition of multi-media should not be based upon current technology. He also makes the very important observation that multiple media are not just built up out of simple parts which can be experimented with separately. Multimedia representations attempt to utilize the "whole mind". Thus single media interface design methodologies may not be appropriate to the multimedia situation.

The PROMISE project

We have been working with multimedia interfaces for the past four years in a large ESPRIT project called PROMISE (PRocess Operators Multimedia Intelligent Support Environment). One of the key goals of the work has been the development of some interface design principles. The process control situation provides some unusual design problems for the multimedia designer which are quite different from the more usual educational or communication situations. For example, a designer can never be really sure when and in what combination "messages" will arrive at the presentation system because of the non-deterministic nature of the process. When something goes wrong, there is usually a surfeit of information competing for limited information transmission channels and therefore media. Thus the designer cannot be sure that the required device will be available at the critical time. Allocating a particular alarm to an audio channel might seem acceptable at the design stage, but in a crisis situation a whole set of audio alarms might arrive simultaneously in the control room.

However, the multi-media approach affords a way out of this scheduling problem, by allowing alternative representations to be used. At the heart of the PROMISE system, therefore, is a multi-media resource manager. This multimedia resource manager uses knowledge about media and the application to resolves multiple media clashes on-line (Alty and McCartney, 1990).

The PROMISE project is based upon two exemplars - a nuclear power plant simulator in the South of Scotland and a chemical plant in The Netherlands. These two plants have been deliberately chosen for the different environments they provide. Work on the nuclear plant has been carried out in the plant simulator, one of the most sophisticated in the field. The work in the chemical plant has been carried out in the control room itself. The simulator allows us to deliberately create error situations and explore the importance of different media in assisting operators to deal with these situations. The real plant offers quite a different challenge - that of longitudinal studies in a real environment where the clock cannot be stopped and unexpected events are intermingled with long periods of trouble free running. In addition to experiments in the simulator and plant, experiments have also been carried out in the Behaviour Laboratory at Loughborough University where conditions can be carefully controlled.

From our expensive laboratory and real-plant studies we have been able to gather together some guidelines or principles for designers. Although the process control area is rather specialised we are convinced that the lessons we have learned from this work are readily generalisable and are therefore applicable to most multimedia design situations.

Some Designer Guidelines

Design principles and guidelines which we have developed in our study of the use of multimedia in process control interfaces (Alty and Bergan 1992) are the following:

The Principle of Telepresence

Multimedia options such as video and audio allow us to regain something which we have lost over the last twenty years of process control interface design - that is - connection with the real world. Gone are the days when the operators could FEEL the plant under their feet, HEAR it in operation, or SEE what was happening and in some cases SMELL the overheating. There are at least two reasons why this telepresence option offered by a multimedia approach is important - the partial elimination of events by proxy, and the preservation of crucial cues.

Firstly, connection to real events is important and is obviously crucial in process control. Our current interfaces tend to offer *events by proxy*. A red light changes to a green light on the control panel to signify the opening of a valve - but did the valve really open ? An event happened to me recently which illustrated this point. The portable Macintosh has an icon which shows that the battery is charging (i.e. connected to the mains). The system was being used in an hotel room when suddenly a message appeared exclaiming that the battery was nearly discharged, however the mains lead was plugged in to a wall socket. On pulling out the cable from the back of the Mac the charging icon disappeared. When the plug was pulled out for the power socket, however, the icon remained on even when there was no connection to the mains. The appearance of the charging icon was an event by proxy. It was triggered by the presence of the plug into the back of the computer, not by the presence of a mains signal. In fact, the problem had arisen because the mains socket in the wall was not switched on. Events by proxy can be a serious source of problems in process control, but the problem is a more general one. With a multimedia approach the operator could actually watch the valve changing state, and hear the valve mechanism in operation.

Secondly, it is often difficult to be precise about exactly what in our environment provides the cues for human decision making. Media (particularly video and audio) carry information which cannot be easily characterised as having a particular use. Good operators simply "know" that there is a problem from hearing sound from the plant, yet they may be unable to define what it is that alerts them. In a similar way the average car driver can sense there is something wrong with a car simply because of a change in the sounds produced whilst running. Operators use cues which designers do not know are there (and in many cases the operators are not conscious of using these cues either). Designers tamper with such information at their peril. In one of our experiments, for example, we found that there was a problem with using the results of a test involved an assessment of the salt level in the process flow. Designers had assumed that a measure of the length of the deposited salt in a narrow tube would be sufficient, whereas it turned out that the colour of the substrate, and even the granulation of the salt, were used in an important way by operators in the evaluation process. Preserving the complete test in a still-colour-picture preserved the vital cues.

The Principle of Measurable Media Differences

Media offer information to operators (or users) to enable them to take decisions. In some cases the decisions will be qualitative ones, whilst in other cases they will be quantitative. It sounds rather obvious to say that the changes in the media must be discernible in an appropriate way by the human observer to be effective. In a number of our laboratory experiments we provided additional sound from the process to help the operators. In a number of cases such sound was found to be irritating and not very useful. The reason was that operators could not use the information in a meaningful way. The differences in the medium could not be related to quantifiable differences in the process, thus control was not possible. An example of this would be representing flow by the sound of a tap. Increases or decreases would be perceived but absolute measurement would not be possible. Thus the use of such a medium would only be appropriate when gross changes were the major cue for action rather than precise changes.

The Goal Principle

We have concluded (along with many others, for example, De Sanctis, 1984) that the main determining factor in media choice is the use to which the information is being put by the recipient. People will argue that this cannot be true. Surely spatial information is always better presented pictorially ? Not necessarily. For example, experiments carried out with weather forecasts have showed that audio information can be superior to visual information in some circumstances even in what appears to be a spatial task. This principle has important implications for multimedia design. What matters in designing a multimedia interface is the nature of the information needed by the users to carry out their tasks. Deciding which medium has characteristics which best transmit such information is then a key aspect of the design process. Ideally, therefore a task analysis (Payne and Green, 1989) should be carried out to ascertain the information needs of the user and the choice of media used will be heavily influenced by those needs.

The information needs of users will, of course depend upon their goals, but it will eventually need to be converted into requirements for particular types of information. For example, operators watching a boiler heating up may have an overall goal of preventing the temperature from rising too quickly. They will therefore be interested not only in absolute temperature values, but in rates of change as well. They might also be interested in the history of the previous five minutes. Alternatively, if rates of change were unimportant, and they may only need to know when a

ERIC

temperature level is reached. In each of these cases the information required has very different characteristics. On the one hand, values, rates of change and historical perspectives are required. On the other merely a change of context. It is clear that using an audio medium for the former task would be inappropriate, but might it be ideal for the second case.

We therefore need a taxonomy for characterising knowledge requirements (generated by user goals). Most work so far has been carried out using graphical media. One starting point could be from the work of Roth and Mattis (1990) who attempted to characterise data in terms of three types - set ordering (whether data is quantitative, ordinal or nominal), domain of membership (time, space, temperature or mass), and coordinate specification (is a point specified spatially, temporally or otherwise). These types can then enter into relationships described in terms of coverage, cardinality and uniqueness. Finally complex relationships can be described. The final section of the paper tries to relate user goals to data characteristics and they provide some useful examples (within graphical media) of how data characteristics govern the effectiveness of the media to present information.

Mackinley (1987) has designed an Automated Presentation Tool. He defines "expressiveness" and "effectiveness" criteria, the former indicating by how far a medium is capable of presenting the data, and the latter indicating how well the medium could present the data. A medium should be chosen which is expressive but not over-expressive. The effectiveness criterion is more difficult to define since it depends partially on the capabilities (and the goals) of the user, but Mackinley develops a presentation scheme for ranking the suitability of different graphical media for the presentation of quantitative, ordinal and nominal data. He suggests a set of primitive languages which can be merged by composition.

Some very interesting work with a direct bearing on the production of a taxonomy is by Hovy and co-workers (see for example Arens, Hovy and Vossers 1992). They identify four types of knowledge which are similar to those mentioned above namely:

- the characteristics of the media
- the nature of the information to be conveyed
- the goals and characteristics of the producer
- the characteristics of the perceiver and the communicative situation

Media have characteristics such as substrate, information carrier, carried item, dimension, temporal endurance etc.., Information is characterised in terms of Dimensionality, Transience, Urgency, Order, Density, Naming and Volume. Producer intentions include teaching, informing, confusing, involving, activating and de-activating goals etc.. Finally, user intentions are concerned with expert student or novice status, interest, opinion, language ability and emotional state.

The Complexity Principle

A question which has never really been answered is "does a multimedia presentation approach assist user understanding a complex situation ? " Most people would feel that the answer ought to be yes, but there is little data to support this viewpoint. We have therefore carried out a set of laboratory experiments to investigate (amongst other things) how far this might be true. We selected a task for this experiment which was closely related to the process control domain and which had considerable scope for multimedia presentations. The task was Crossman's Waterbath (Crossman & Cooke, 1974). It has been used on numerous occasions (Moray et al, 1986, Moray and Rotenberg, 1989) since it was first used by Crossman and Cooke to show operator progression from a second order controller to exhibiting open-loop control. The Waterbath Task, illustrated in Figure 1, is a simulated thermal hydraulic system which consists of a single tank (in the variation used for this experiment). An inflow and an outflow pipe are connected to the tank.

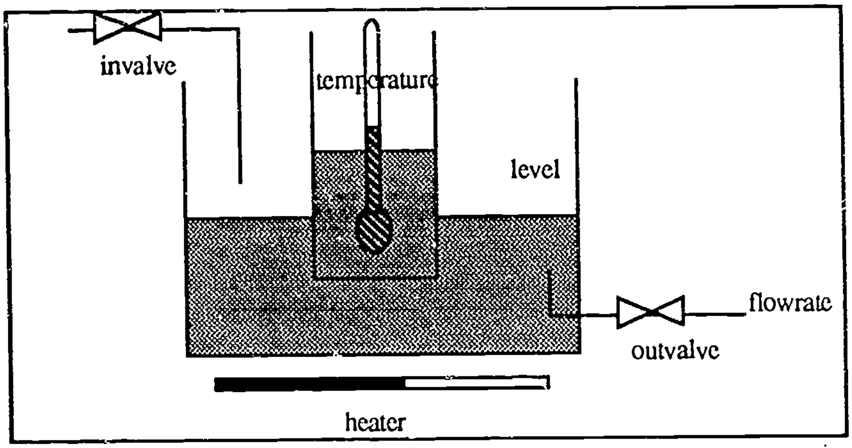


Figure 1 Crossman's Waterbath

There is a valve on each pipe (in-valve, out-valve) which may be used to regulate the flow. The heater is situated immediately underneath the tank. Inside the tank is an insulated container containing a fixed amount of water and a thermometer. There are also imaginary sensors to measure the level of water in the tank and the rate of flow out of the tank. There are three control variables: *in-valve*, *out-valve* and *heater*. The state of the process is indicated by the three variables *level* (in millimetres), *flow-rate* (millilitres per second) and *temperature* (°C).

The tasks which the subjects had to solve, involved achieving new steady state process states from a steady state starting point. The target state was defined as a range within which each of the system variables had to lie after stabilisation of the system. Subjects were placed in front of a PROMISE terminal and their actions were recorded on two television cameras. Sound and speech activity were also recorded. In addition, some gesture information was recorded (as it happened) on the video tape by a hidden observer. More than 50 subjects participated in the experiment. They were mainly undergraduate students in Computer Science and were also mainly male. The few females who took part were balanced across the experimental conditions. Each session lasted about 2 1/2 hours including a debriefing interview. The full set of results have been reported elsewhere (Alty, Bergan, Craufurd and Dolphin 1993)

Different media were used in the various interfaces. For example, the TEXT-ONLY interface showed the three current text values of the three output variables, and the user could alter numeric values of the input variables. In the GRAPHICS-ONLY interface a comprehensive graphic (in colour) was shown to the user (very similar to Figure 1) with output variables being shown by sliders. Other interfaces had visual or verbal warnings and sound (the water flow) or combinations. Finally, we provided some unmediated video output.

During the experiments users were tested for their comprehension of key concepts and this comprehension was measured as the percentage of statements made which were correct. The results for the six main concepts are displayed in Figures 2 (a) and (b). It can be seen that comprehension varies from nearly 100% (whatever the media) for temperature, heater and in-valve to less than 50% for flow-rate and out-valve, with level comprehension being about 80%.

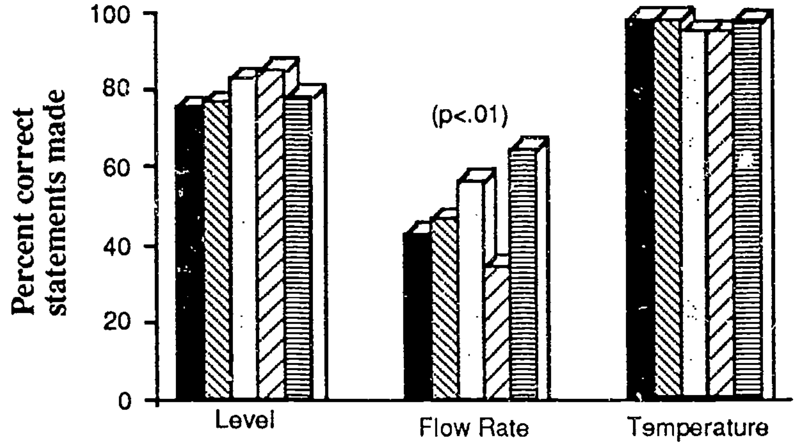


Figure 2 (a) Multimedia Presentation and its effects on Comprehension

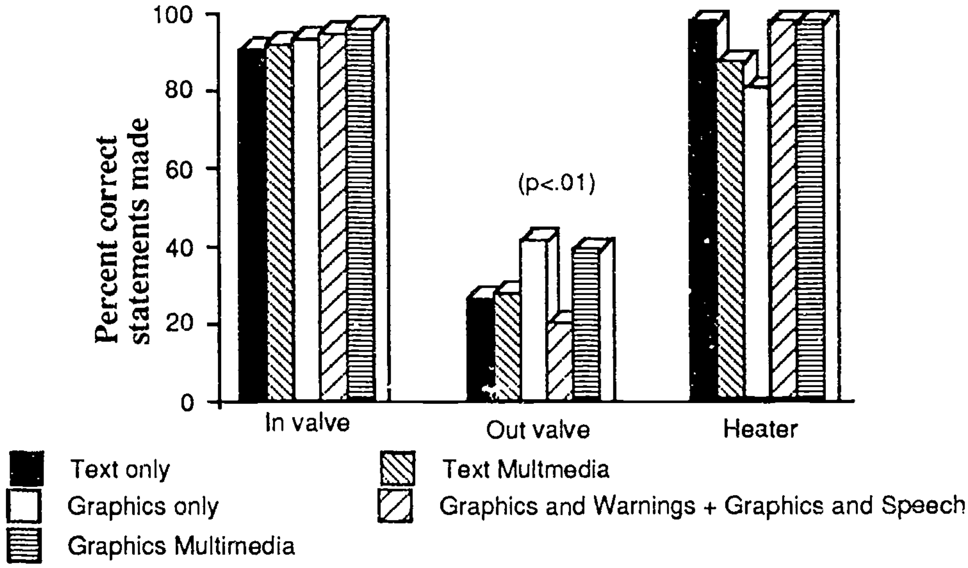


Figure 2(b) Multimedia Presentation and its effect on Comprehension

It appears that the usefulness of different media in presentation situations is closely related to the complexity of the idea being conveyed and the capabilities of the perceiver. For example, a simple concept (such as increasing heater output leads to a rise in temperature in a water bath) is so obvious a relationship for most perceivers that almost any medium would successfully convey it. Thus we see comprehension values of nearly 100% with little variation across media. On the other hand, trying to communicate a more difficult concept (for example, Einstein's Tensor Calculus) might be so difficult to grasp that whatever combinations of media are used the presentation will fail for all except the competent mathematician as a perceiver. Between these extremes there will be concepts whose comprehension is rather difficult (for example the out-flow relationship which is difficult to grasp because of its dependency on level) and the interesting result is that comprehension is apparently influenced by medium choice in this situation. This is a useful result because it corresponds to a very common situation - that when operators or users are forced into a situation where their competence is stretched, such as fault finding in the process control situation.

The Principle of Apparent Redundancy

It appears that in the multimedia situation - more is better. Human beings seem to prefer parallel sets of redundant information rather than unique sets of accurate information. We are using the term redundant to mean "useful but not strictly necessary". For example, providing spoken text alongside written text can be very beneficial to comprehension, but except in the case where the operator might be moving about the room, this might be thought of as being redundant information. It appears that the more human beings use redundant (but individually useful) information the nearer they get to exploiting the "whole" mind. An overhead colour shot of a snooker player sinking a shot is enough to tell an observer what is going on, however the sound, although apparently redundant, does aid comprehension. It gives an alternative, reassuring confirmation of the event. This is an important use of multiple media.

The Principle of Operator Choice

The choice of the "best" medium depends upon the operators' goals (The Goal Principle). However it is not always easy to determine the full range of operator goals some of which are often hidden. It is therefore important to allow operators some media flexibility. This will allow them to exploit the interface and will provide useful feedback on hidden operator goals. In the PROMISE system we allow operators to express their preferences and, if possible, honour them.

The Intrusion Principle

Some media are more powerful than others at helping operators to solve certain goals. An example of an intrusive medium would be sound when used to alert operators of alarm conditions. Such media can be very successful but can impair performance on other parallel goals. Therefore, only use intrusive media when the intrusion is required to solve a higher level goal and you are prepared to risk the loss of concentration on other lower level goals. Be sure to re-introduce the neglected goals when the emergency is over.

A switch of medium can be very effective in forcing a context switch on an operator. Sound is often used for this purpose, but other media switches can work as well. For example, a visual switch in a sound intensive environment can be very effective (provide the operators attention to the screen can be guaranteed). Media switching can also shed new light in problem solving.

The Principle of Media Synchronisation

Some media work extremely well if synchronised together. The obvious example is, of course, video and audio in film or television and the synchronisation is essential to carry over a notion of reality. However there are other useful synchronisation possibilities - for example, animated graphics and text overlay (or sound commentary), feeling in a data glove and visual effect (i.e. firing a projectile in a computer game), or visual movement in a virtual reality world synchronised with hand movement. The important point here is that the synchronisation may not just be creating "real-world" like conditions, it may also be minimising the overload on short-term memory.

Conclusions

These principles are, of course, only a stepping stone to a full set of design principles for the design of multimedia interfaces. No doubt more will emerge as work proceeds. We need considerable effort in matching knowledge characteristics to media characteristics, and in recognising why different media give designers an advantage in presentation.

Acknowledgement

The authors would like to acknowledge the real contribution made by the other partners in the PROMISE project. The partners in addition to LUTCHI are: Scottish Power in Glasgow, Scotland, DOW Benelux at Terneuzen, The Netherlands, EXIS, a software company based in Pisa, Italy, the University of Leuven in Belgium and the Work Research Centre in Dublin, Ireland. The PROMISE project is supported by the European Commission as project 2397 in ESPRIT-2 and we gratefully acknowledge this financial support.

References

Alty, J.L., (1991), "Multimedia: What is it and how do we exploit it ?", Keynote Address to HCI'91, (ed. D. Diaper and R. Winder), pp 31-44,Cambridge University Press.
Alty J.L., and McCartney, C.D., (1992), "Managing Multimedia Resources in Process Control: Problems and Solutions", in Multimedia: Systems, Interactions and Applications, (L. Kjelldahl ed.), pp 293 - 306, Springer-Verlag, Berlin.
Alty J.L., and Bergan, M., (1992), "The Design of Multimedia Interfaces for Process Control", Proc. 5th IFIP/IFAC/IFORS/IEA Conf. on Man-Machine Systems, The Analysis, Design and Evaluation of Man-Machine Systems, (Stassen, H., ed.), The Hague, pp 249 - 255.
Alty J.L., Bergan, M., Craufurd, P., and Dolphin, C., (1993), "Experiments using Multimedia Interfaces in Process Control: Some Initial Results", in Computer Graphics, to appear.
Crossman, E. R., and Cooke, F. W., (1974), "Manual Control of Slow Response Systems", in The Human Operator in Process Control, (E. Edwards and F. Lees eds.), London: Taylor and Francis, pp 51 - 64
Marmollin H., (1992), "Multimedia from the Perspectives of Psychology", in Multimedia: Systems, Interactions and Applications, (L. Kjelldahl, ed.), pp 39 - 52, Springer-Verlag, Berlin.
DeSanctis, G., (1984), "Computer Graphics as Decision Aids", Decision Sciences, Vol 15, pp 463 - 487.
Arens, Y., Hovy, E., and Vossers, M., (1992), "On the Knowledge Underlying Multimedia Presentations", in Intelligent Multimedia Interfaces, (ed. Maybury, M.) AAAI press, pp 1 - 25.
MacKinley, J., (1986), "Automating the Design of Graphical Presentations of Relational Information", ACM Trans. on Graphics, Vol 5,No 2, pp 110-141.
Moray, N., Lootsteen, and P., Pajak, J., (1986), "The Acquisition of Process Control Skills", IEEE Transactions Systems, Man and Cybernetics, SMC-16 pp 497-504.
Moray, N., and Rotenberg, I., (1989), "Fault management in Process Control: Eye Movement and Action", Ergonomics 32(11) pp 1319-1342.
Payne, S., and Green, T., (1989), "Task-Action Grammars: The Model and its Developments", in Task Analysis for Human Computer Interaction, D.a. Norman and S.W. Draper, (eds.), Lawrence Erlbaum Associates.
Roth, F.R., and Mattis, J., (1990), "Data Characterisation for Intelligent Graphics Presentation", Proceedings of CHI'90, pp 193 - 200.

Multimedia: A Revolutionary Tool To Enhance Teaching And Learning Of Structural Steel Design

Abbas Aminmansour
Architectural Engineering Department
The Pennsylvania State University
University Park, PA 16802 USA

Why Interactive Multimedia Technology?

Interactive multimedia technology offers several unique advantages over traditional methods of education and training. One such advantage is the module's involvement of the user. The user may control the speed and direction of the lesson being presented by menu selections or by responding to questions. This interaction offers individualized, hands-on learning, and self-paced instruction which contribute to better comprehension and retention than traditional methods.

Interactive multimedia applications educate learners the way most people acquire information -- through active learning. A right combination of this technology, instructional design, expertise in a subject matter, and creative minds could result in tools which not only improve learning and retention of the subject, but they do so in a more effective and entertaining way.

These applications also allow random access, immediate feedback and automatic or manual review. They demand viewer attention, turning passive observers into active participants, totally involved in the learning process. Further, by applying creative techniques, one may produce tools that are fun to work with and maintain a high level of interest in the user. Finally, multimedia modules are easy to use and are often more economical compared to other alternatives.

The SteelDEM Project

A project is underway at The Pennsylvania State University to develop Steel Design Educational Module (SteelDEM), an interactive multimedia computer application for teaching/learning of design and development of steel structures. The module will cover different structural steel members, their behavior, limit states, and design. The effects of different variables on the physical behavior and strength of members will be studied through animations of different situations and discussions of design specifications, formulae and plots.

The module will allow users to create and interactively solve problems with "real world" constraints and limitations. These conditions may include such constraints as availability of structural shapes and steel types, common member sizes, interaction of different members and the effect of member sizes on ease and economy of connections between them. Within each section, the module will contain exercises of different difficulty levels to ensure a better understanding of the subject matter. SteelDEM will be developed based on the latest Load and Resistance Factor Design (LRFD) of steel structures, adopted by the American Institute of Steel Construction.

SteelDEM will be developed in two phases. Phase I will lead to development of a version with all major features of the final module but with a limited amount of contents. This version, expected to be ready by summer of 1993, will be tested and evaluated at a number of schools. Phase II of the project will begin immediately following the completion of phase I, and will continue for a period of about three years. The second phase of the project will result in the completion of SteelDEM including all the contents and features explained in this paper.

The Module: Its Contents and Features

SteelDEM is structured similar to a typical textbook. It consist of a number of major parts or "chapters", each of which is divided into sections and subsections as necessary. Easy navigation systems allow users to move between sections quickly and with ease. SteelDEM is divided into the following major parts:

1. Tutorial;
2. Structural Steel: Production, Fabrication, and Erection;
3. AISC Manual;
4. Tension Members;
5. Columns;
6. Beams;
7. Beam-Columns;
8. Fasteners/Connections;

Following is a brief description of some of these components.

The **Tutorial** will consist of instructions as well as examples of how to use the module and its features. It may be accessed at anytime from within the module.

Structural Steel: Production, Fabrication, and Erection will include about forty five minutes of menu controlled video, computer graphics and animations on how structural steel is produced, fabricated, and erected. In particular, design considerations for more efficient and economical design of steel structures will be addressed using graphics, animations and footage of "real world" examples as well as interviews with educators and other professionals such as consulting engineers, architects, fabricators, and contractors. This part will also have an index for quick access to desired topics.

AISC Manual will include pertinent parts of the *AISC Manual of Steel Construction (LRFD)* such as the data base of structural shapes, specifications, as well as other necessary tables, charts, and design aids. Although this part is included as a separate "chapter", it may be accessed from any part of the module. AISC Manual will have a menu system of its own for easy navigation. In addition, any pertinent parts of the manual will be grouped and made accessible in the problem solving screens through buttons. All charts and tables are reproduced in the same format as they appear in the AISC Manual to avoid confusion by users.

Chapters four through eight address material typically covered in similarly titled chapters of a structural steed design textbook. Each of the chapters will be broken into sections. As an example, the chapter on columns, currently under development, is divided into the following sections.

1. Concepts and Definitions
2. Buckling
3. Facters Affecting Column Strength
4. AISC Specifications and Formulas
5. Analysis of Columns
6. Design of Columns

The first two sections cover the basic concepts, definitions and buckling of compression members. An assortment of color pictures from real structural components, full motion video, computer graphics, and animations are used in these and other sections to present the content more clearly. Section three discusses different factors affecting the physical behavior and strength of columns such as laterally unbraced length, effective length factor/end conditions, cross sectional properties, and material properties. The relationships among different variables are studied through visual aids as well as plots and formulae.

The section titled AISC Specifications and Formulas will discuss engineering specifications, general concepts, and procedures involved in solving a variety of problems utilizing design aids included in the AISC Manual. As with any other section, the user may decide to skip this section if (s)he so desires.

Sections five and six of this chapter cover analysis and design of columns. Several methods of analysis and design of columns will be introduced. Each section will include interactive numerical exercises and examples as well. Each section also allow users to solve problems pre-determined by their instructors or created by themselves. SteelDEM's problem solving capability features three difficulty levels. In level I, intended for users new to the subject, the actions taken by the user will be closely monitored while hints and instructions help guide users through a problem.

In level II, intended f . intermediate level users, the user is supplied with a list of possible tasks for solving a problem. The user must decide which tasks are appropriate and in what order. Within each task, the user does all the work and the computer acts as a facilitator. In level III, the advanced level, the user will only have access to the specifications and design aids. (S)he must decide what to do and which formulae and design aids to use. At this level, users may be restricted to certain constrains to make the problems more challenging. Different methods of problem solving will be available as options, giving users more flexibility. In all levels, users will have on line access to the AISC Manual and their work will be automatically checked for compliance with the latest available AISC specifications.

25

Other Features

Nine buttons remain as permanent parts of all screens in the module. These buttons are: Menu, AISC Manual, Glossary, Notebook, Set Bookmark, Navigate, Calculate, Help, Index, and Exit. The user may return to the original location in the tutorial after using features accessed by any of these buttons. Most screens have other features and buttons which are specifically included for those screens.

The **Menu** button displays a hierarchy of all chapters and their sections. It also allows the user to navigate quickly through the tutorial and jump from one chapter to another or move between sections within the same chapter without going back to the main selections. This feature also identifies the present location of the user in the module and identifies chapters and/or sections that have been visited by the user during a session.

The **AISC Manual** will give users quick and on-line access to pertinent parts of the *AISC Manual of Steel Construction (LRFD)*. Information available in this part may be automatically transferred to another part of the module. In solving problems, all necessary information may be obtained directly from the *AISC Manual* by SteelDEM without the user having to go to that part.

The **Glossary** consists of a compilation of the terminologies and symbols used in the module, including the terms and symbols used in the *AISC Manual* along with their definitions and units, if any. Some terms and symbols in the Glossary are accompanied by video, graphics, or animations to better illustrate what the selection represents. Further, if the selection is related to other words or symbols, those words and symbols appear at the end of the definition as "hot words". The user may click on any of those words in which case (s)he will be taken to another part of the Glossary where the related term is listed. The Glossary has a menu system for quick access to its contents.

Notebook gives users access to an "electronic notebook" which may be used to record remarks or notes for future reference and/or to print. The **Set Bookmark** button allows users to set electronic marks at any point in the module for easy access in the future. Each bookmark will be labeled for identification. The user may save his/her bookmarks on a floppy disc for use in future sessions.

The **Navigate** button provides users with a list of bookmarks set earlier. Selecting a bookmark title in the list will take the user back to that location. The **Index** button gives users the capability to go to pertinent part(s) of the module for further study of a subject. The **Calculate** button brings up a scientific calculator for use in certain exercises or for any other purpose. Although the computer is capable of carrying out all necessary calculations very quickly, SteelDEM is developed with the idea of having the user carry out some calculations for better understanding and appreciation of certain concepts.

The **Help** button provides users with on line help on how to use the module or to explain certain features. The **Exit** button allows the user to close the module at any time with the option of saving the results of that session such as problems solved or bookmarks set.

Other SteelDEM features include the capability to electronically evaluate students' work and save their scores, print or save each student's work, and record a history of the student's use for future review by the student or the instructor.

Because of its educational nature, SteelDEM is being developed with the intent to incorporate maximum interaction with users. It is designed to require the user to be very involved through exercises, taking actions and making decisions at different stages of a session.

The module also allows users to select different kinds of connections between members and evaluate different alternatives in terms of strength, economy and ease of design, fabrication, and erection. This feature includes a library of animated parts of connection which may be assembled and studied.

SteelDEM could be placed on electronic networks and made accessible to users from a variety of locations including classrooms, laboratories, or residences. The module will be designed with flexibility to allow future updates and modifications due to changes in technology and/or contents. It will also be possible to expand the module in the future to include more advanced topics.

Instructors may wish to set up local area networks and keep records of students' use of the module and evaluate their responses to questions. They may also wish to store sets of problems which could be printed as assignments at the end of a session. Additionally, reading and problem assignments may be suggested to students who may have shown weaknesses in certain areas.

SteelDEM will not only enhance engineering students' understanding of structural steel design, but will also familiarize them with this new technology which they will very likely face in their future careers.

System Requirements

The recommended minimum system configuration for using SteelDEM is as follows:

1. A 386 (or higher) IBM PS/2 or compatible computer with one floppy disk drive, XGA/Super VGA display, mouse, and Microsoft Windows;

2. Four megabytes of memory;

3. A CD-ROM drive;

5. Network connection and appropriate hardware and software needed for networking (optional).

Depending on the state of technology at the time of distribution, SteelDEM may be available on more than one medium when completed.

Project Advisory Board

In order to ensure a high quality in design, development and contents of the module, an advisory board of twelve prominent persons in education, professional practice, and computer technology has been assembled for the project. Members of the Board receive periodic reports on the progress and contents of the module and provide valuable input from their own perspective. Members of the Board include faculty from the following universities: University of Illinois (Urbana), University of California (Berkeley), University of Wisconsin (Madison), Massachusetts Institute of Technology, Cornell University, and The Pennsylvania State University. A number of others specializing in computer technology, instructional design, and structural engineering are also members of the Advisory Board.

Testing and Evaluation

SteelDEM will be tested and evaluated as follows.

1. During production -- beginning in Fall Semester 1993, parts of the module already developed, will be tested and evaluated at several universities. This process continues through the entire production.

2. After production -- input/feedback will be solicited from users. Future upgrades will incorporate the necessary changes and features communicated by faculty and student users.

3. A statistically designed study will be conducted to evaluate the impact of the module on students' learning as well as any changes in their motivation or interests in the field as a result of using SteelDEM.

Acknowledgments

Support for the SteelDEM project is provided by the American Institute of Steel Construction (AISC), International Business Machines (IBM), National Science Foundation (NSF) and The Pennsylvania State University. Support from NSF is provided through the Engineering Coalition of Schools for Excellence in Education and Leadership (ECSEL). The Pennsylvania State University supports the project through its College of Engineering, Department of Architectural Engineering, Computer Based Educational Laboratory, and University Photo/Graphics Services.

ERIC

Logical Markup For Hypermedia Documents: The TRAIN-System

Friedrich Augenstein, Thomas Ottmann, Jürgen Schöning
Institut für Informatik, Universität Freiburg
Rheinstrasse 10-12, D-7800 Freiburg, Germany

1 System architecture

Our experience has shown that the ever increasing functionality of modern authoring systems for developing computer aided instruction material does not suffice to guarantee high quality courseware. In particular unexperienced and casual authors of CAI-material need tools which incorporate the knowledge of experienced courseware designers, cognitive scientists and pedagogues [AJSW][MMO][Ott].

We will report about the design and implementation of a prototype-authoring system called TRAIN(type and rule directed authoring) which enables a clear distinction between the tasks and responsibility of the author on the one hand and different kinds of experts in courseware development on the other.

The TRAIN-System allows a strict separation of the logical content-structure of a course from its formal layout-structure. This is achieved by following well established principles underlying modern document preparation systems like LaTeX [Lam]. We will introduce the notion of *type* and *type-directed* authoring and presentation. The author only specifies what type a special part of a multimedia document should have. The layout of the objects of a given type is described in style sheets which are established by professional designers. The linking scheme of information units is described by pedagogues.

As a result of the editing process using TRAIN we obtain a symbolic description of the multimedia document with a logical markup. The markup language which we use resembles the Standard Generalized Markup Language (SGML)[Bry] used in document processing. Types specified in this language are ultimately reduced to basic types (e.g., #PCDATA). Types may have attributes which obtain specific values on any instantiation; they may also have procedural elements, called *scripts*, which implement different ways of editing or presenting objects of the given type.

There are other approaches based on SGML for modelling multimedia documents. The ISO published a standard called HyTime [ISO]. In comparison to HyTime, our system is much more courseware-oriented and provides the expert with types and attributes that he can use directly to lay down his experience in types which will then support the author.

The TRAIN-System consists of two major parts, the *editing system* and the *presentation system*. The overall structure of a course is defined by a Document Type Definition (DTD). It specifies the class of all possible specific structures and directs the editing process. The editing process is supported by type-specific editing rules. As a result of this type-directed authoring we obtain a multimedia document with a logical markup. A precompiler expands this document using a given style sheet into a fully expanded version which contains only

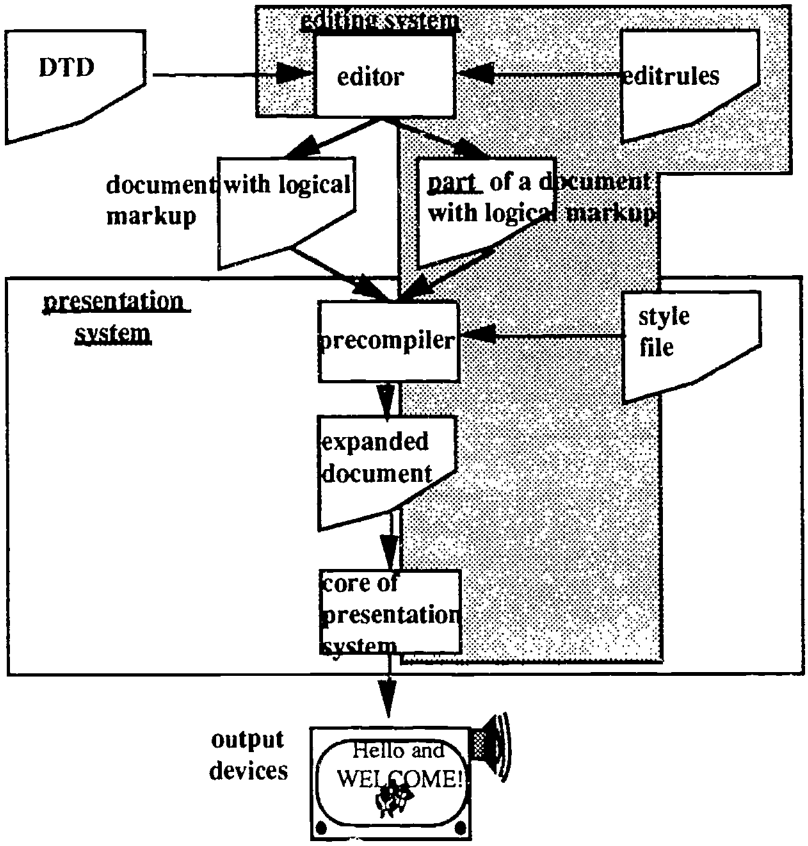


Figure 1: The architecture of the TRAIN-System

base-types and base-attributes. They are interpreted by the core of the presentation system. The editing system, too, uses units of the presentation system to make parts of the document visible and audible for the author while editing the course. Figure 1 shows the overall architecture of our system.

2 Types

In TRAIN, types are defined in an SGML-like language. A DTD for the titlepage of an electronic course might look as follows:

```
<!ELEMENT Title (Titlestring, Author, Institution, session_time)>
<!ELEMENT Titlestring (#PCDATA)>
<!ELEMENT Author (#PCDATA)>
<!ELEMENT Institution (#PCDATA)>
<!ELEMENT session_time (#PCDATA)>
```

The type `Title` consists of four subtypes which are strings (#PCDATA). The following document may serve as an example of a document generated by the above mentioned DTD:

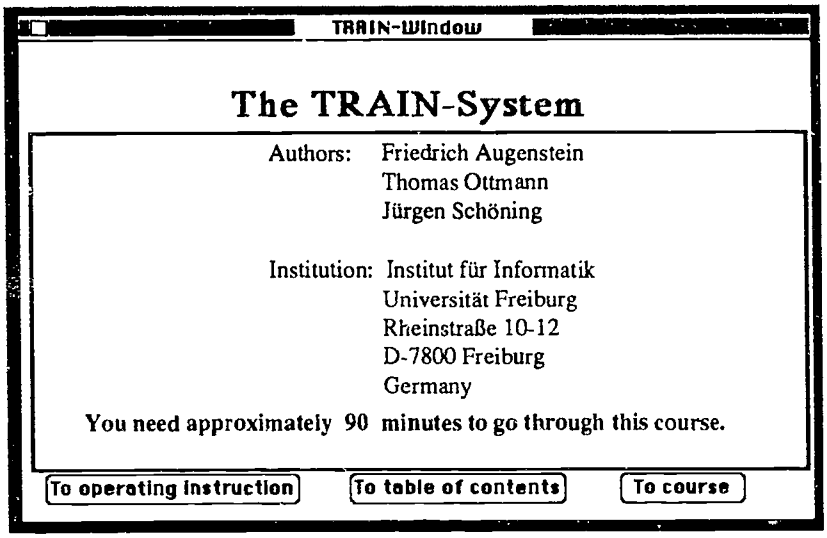


Figure 2: An expert-defined layout of a course-titlepage

```
<Title>
      <Titlestring> The TRAIN-System </Titlestring>
      <Author> Friedrich Augenstein, Thomas Ottmann, Jürgen Schöning
      </Author>
      <Institution> Institut für Informatik ...</Institution>
      <session_time> 90 </session_time>
</Title>
```

The layout of a document of type `Title` is described in a style sheet by a design expert and might lead to a presentation like the one in figure 2. The expert has defined the position and style of the strings for `Titlestring`, `Author`, etc., he has created links to other parts of the document (e.g., to table of contents) and has filled in author's content in predefined strings. ("You need approximately `session_time=` *90* minutes to go through this course.")

It is obvious that a document with a tree-like structure is created. In our example the root of the tree would be of type `Title`. The sons of this root would be an object of type `Titlestring`, an object of type `Author`, an object of type `Institution`, and an object of type `session_time`. An object consists of a type name, an attribute list, and data (param string). A next-pointer points to the next element in the tree on the same level. A son-pointer points to the successor in the tree.

We distinguish three different classes of types, screen types, structure types, and interaction types. Each class consists of some basic types and complex structured types which can be defined by using the SGML-like language.

Screen types are the types of objects visible on a (dynamic) screen. As base-types, we have implemented the standard types `Picture`, `Line`, `Rectangle`, `RoundRectangle`, `Oval`, `Word` for graphical primitives and `Text` and `Phrase` for text primitives with base-attributes specifying color, line size, etc. Multimedia types like audio, video or animation sequences can be defined in the stylefile by using a HyperCard-external-command (XCMDs) function. Overlaying, timed sequencing, and erasing of several objects are realized by using

31

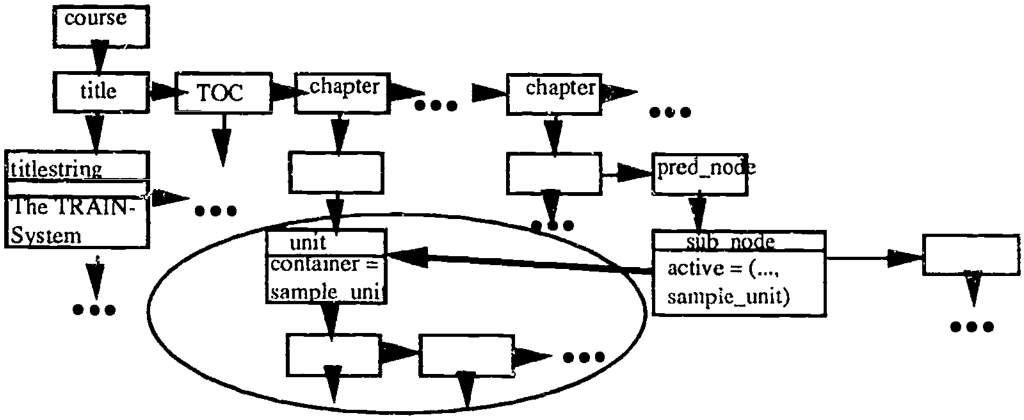


Figure 3: A document tree with "hypergraph" parts

a base-attribute `Dias` and a base-type **`eraser`**.

We cannot only structure documents in a hierarchical, tree-like manner. We can also consider a whole subtree of the document tree as a single unit using the base-attribute **`container`** and set arbitrary links between those units. A **`container`** attribute can be used by experts to define such **structure types**. It specifies an identifier for a node in the document tree and, thus, implicitly for the whole subtree rooted at this node; the **`active`** attribute described below specifies the reference that points to a certain identifier. Any fully expanded document has therefore the structure of a multilevel hypergraph (of the class specified by the DTD) with hypernodes, i.e., sets of nodes, and hyperedges between hypernodes as in the hypertext model introduced by Frank Tompa in [Tom]. Navigation through a document now means to follow tree-edges or hyperedges.

By specifying **interaction types**, user interaction can also be lifted to a "logical" level. The author may, for example, use a "next" command but must not specify whether following "next" means a key-press, a mouse-click or a selection from a pull-down menu. This is layed down in the style file in a consistent manner. The base-attribute for interaction types is **`active`**. It specifies the kind of interaction and the label to jump to when activated. The label is set by a **`container`** attribute as described above.

3 Hypergraphs

As mentioned before, a document is internally represented in a tree-like manner. In figure 3 an example of such a document tree is shown. A document part of type **`sub_node`** in the lower right corner has an **`active`** attribute with the parameter **`sample_unit`**. This corresponds to an hyperedge to the node of type **`unit`** whose **`container`** attribute is marked with the same parameter. This way the **`container`** and **`active`** attributes impose a hypergraph structure on top of the hierarchical tree-like document structure. Figure 4 is a hypergraph view of the "hypergraph" document parts shown in figure 3.

ERIC

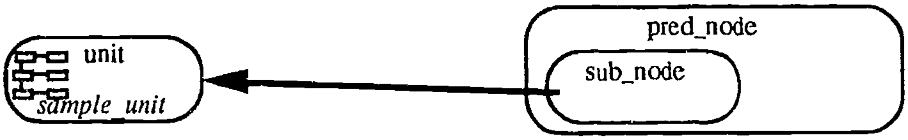


Figure 4: A hypergraph view of the document tree parts in figure 3

4 Simple examples of style rules

For each type which is not a base-type the style file contains expert rules of how to present objects of the respective type. The style file is structured in the same way as a document with logical markup and can thus be defined by a DTD. For that purpose, the style file may contain the procedural types **repeat, get, if, proc** and procedural elements, called "scripts" (introduced by a #-sign). As a simple example for a screen type defined by an expert we declare a type `FrameTitle` which can be used to provide a (dynamic) screen (a "frame") with a title. The DTD contains the following specification:

```
<!ELEMENT  FrameTitle   (#PCDATA)>
```

When the syntax-directed editing process reaches the stage of producing a frametitle, the author will be requested to input some text as a title. An expert designer may have specified the following presentation mode in a style file (*not* accessible to the author):

```
<stylerule element=FrameTitle
        run=1 >
  <Shape  curTextStyle=(Helvetica,outline,14)
          curRGB=(60000,0,30000)
          curbackRGB=(60000,60000,60000)>
     <Word>(500,2200,0,0,#getData(cE))</Word>
   </Shape>
</stylerule>
```

It requires the expansion of the structured document edited by the author on the first run of the (pre-)compiler (`run=1`). The expanded document will consist of a graphic primitive (`Shape`) with assignments of specific values to base-attributes specifying text-style, foreground and background colors (here: the red, green, and blue parts of the color). Then an object of type `word` will appear with reference point at point (500,2200) in a coordinate system fixed by the expert; the external part of the object will be the text input of the author. (`#getData(cE)` means: get the author's `#PCDATA`-input of the current element (which is of type `FrameTitle`))

Structure and interaction types can be specified by similar means. The author specifies the course structure on a logical level; any details of how to follow the linking structure (like pressing a key, clicking a button, etc.) has been specified in a style rule by the expert.

Defining and using style rules is a very powerful mechanism. As examples we have created style rules for automatic generation of an interactive table of contents, for bar charts, and for several graphic types. (e.g., a figure containing a text and a graphic in a predefined arrangement.)

5 Syntax-directed editing

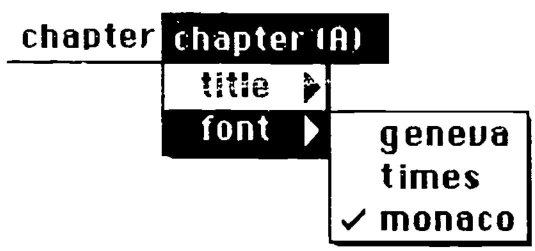
The DTD guides the menu-driven editing process. The author is forced to follow the predefined structure and to use the logical commands. This is achieved by coupling the menu-structure and the DTD. For every type the editor generates a menu. The menu name is the name of the type. The names of the menu items are the names of the subtypes. Consider the following type definitions in a DTD:

```
<!ELEMENT lesson         (introduction, main, consolidation)>
<!ELEMENT introduction   (#PCDATA)>
<!ELEMENT main           (chapter+)>
<!ELEMENT chapter        (#PCDATA)>
<!ATTLIST chapter        title    (CDATA)
                         font     (geneva | times | monaco)    monaco>
<!ELEMENT consolidation  (#PCDATA)>
```

Objects of type **lesson** can be edited by editing objects of its three subtypes. The editing of an object of type **introduction** and **consolidation** is managed by inserting a string (#PCDATA), while objects of type **main** are edited by editing one or several (+) objects of type **chapter**. The type **chapter** also requires a string. It also has some attributes defined in the <!ATTLIST ...> statement. Here the attribute **title** requires a string (CDATA) and the attribute **font** one out of three values **geneva**, **times** or **monaco**. **monaco** is set as a default value.

In our example the DTD would create a menu for type **lesson** (left). Only the item which the author is allowed to choose next is emphasized. The others are disabled and can only be chosen after the first item has been edited. Having done so, the menu will look like the picture on the right. The first item is marked as edited and the next item is emphasized. Editing a sequence of chapters is achieved by selecting from a corresponding sequence of menu items.

If an object is edited which is of a type that requires only #PCDATA, a window will open where the author can insert text. If a type has attributes, they will be displayed in an extra menu. If the attribute is CDATA, the same window as for inserting #PCDATA-strings will open to insert text. If there is a list of different choices, the actual choice is marked as the following picture shows.



ERIC

In addition, we associate respective editing methods to certain types which we have called editing-rules. They provide the author with a comfortable environment to edit objects of a given type. They automatically assure that all type-restrictions will be obeyed, and, thus will generate a structured part of a complex multimedia object with logical markup. As an example consider the editing-rule for creating a rectangle. A DTD defines it as

```
<!ELEMENT   Rectangle   (#PCDATA)>
```

After the editing process the string #PCDATA must contain the coordinates of the rectangle. Thus the **editrule** for type **Rectangle** must provide the author with a tool for specifying the four coordinates. The editrule looks as follows:

```
-- The editrule for RECTANGLE should not generate a menu item; --
-- so set the menu attribute to "false". --
<editrule  element=Rectangle
           menu=false>
-- Now there are two editing procedures; their functionality is
-- implemented in the editing system; #EditPoint specifies a point
-- determined by a mouse-click; #EditRect specifies the rectangle;
-- the rectangle is opened from the point specified by #EditPoint
-- and then drawn using a "rubberband" while the mouse is moved;
    <proc> #EditPoint
           #EditRect
    </proc>
-- #EditRect specifies four global variables; these variables are now --
-- used by the set type; this type sets the #PCDATA of the current --
-- object which is of type Rectangle; the four variables are put --
-- in the #PCDATA of the current Rectangle. --
    <set>(RectLeft, RectTop, RectRight, RectBottom)</set>
</editrule>
```

The editing process carried out by the author might result in the following instance of a document part:

```
<Rectangle>(10,20,9000,8000)</Rectangle>
```

6 The link editor

Another part of the editing system is a link editor which is currently being developed. So far, links can be set by inserting CDATA-strings in attributes as described above. Our aim is to enable the author to establish these links graphically so that the units will be specified by icons and the links will be drawn by the author as lines connecting these icons. This process will be visualized as a hypergraph on the screen. Among others, solutions for the following questions have to be found: What kind of information units (icons) have to be distinguished? How can they be visualized? How can the different hierarchical levels of the hypergraph be made visible? Are there different kinds of links? How can we distinguish between them? How can a link be visualized that connects icons of different hierarchical levels? What kind of view should the author be provided with? Should global information be visible? (e.g., "You see 10 % of all icons and 5 % of all links.")

7 Answer analysis

An important feature of courseware is the question/answer-interaction. Answer judging can appear in different forms: clicking somewhere on the screen, making a choice from a pull-down menu, typing in some text, etc. In our system an expert can define these kinds of interaction in a style file by using the base-attribute `active` and establishing composed types, e.g., a multiple-choice by defining click-fields and corresponding links to feedback units.

We have concentrated in particular on classifying text answers as "wrong" or "right". The problem here is that many different answers can considered to be correct. We restrict ourselves to a purely syntactic approach for judging text answers and avoid knowledge-based or natural-language interface systems. This is the common approach in modern authoring systems. A simple comparison of input and pattern strings is not enough, because users make typing errors, use synonymous words or may not follow an intended word order. Therefore, in TRAIN the author is provided with a powerful language to describe a class of "right" answers.

The language exceeds the expressive power of corresponding languages in systems like AUTOOL [GM] or Authorware Professional [Aut]. It allows to specify positive and negative occurrences of words (with a given length) and phrases (consisting of several words, "don't care"-symbols, etc.) to require the occurrence of words and phrases in a given order, and to check for approximate occurrences, i.e., searching for words and phrases within a text with a given edit distance from the "correct" answer.

New algorithms for approximate string matching have been developed and implemented which resemble the Shift/OR-algorithm of [WM] and the dynamic programming algorithm of [Ukk].

8 Implementation

Our prototype implementation of the TRAIN-System has reached the stage where the presentation system is fully operating. The editing system is incrementally improved by adding more and more editing modules. We have to design a large variety of style files in order to increase the comfort to suit many different authors. So far we have designed a few style files for demonstration purposes and some short courses (on topology, chess-end games), for which we [illegible] show different presentation modes by exchanging style files. At present we define [illegible] style files in cooperation with pedagogues of our university.

It [illegible] r goal to create a commercial product for widespread use. However, we want to sh[illegible] t type-directed authoring may lead to computer-aided instruction material of a high q[illegible]y.

References

[AJSW] F. Augenstein, Th. Jechle, J. Schöning, and A. Winter. Design- und Didaktikhilfen als vorimplementierte Modelle für den Lernsoftware-Entwurf. In P. Gorny, editor, *Informatik und Schule 1991*, pages 225-234, Springer-Verlag, Informatik-Fachbericht Nr. 292, Oldenburg, Germany, 1991.

[Aut] *Authorware Professional Reference Manual.* Authorware Inc., Minneapolis MN, 1990.

ERIC

[Bry] M. Bryan. *SGML: an author's guide to the Standard Generalized Markup Language.* Addison-Wesley Publishing Company, 1988.

[GM] J. Garratt and H. Maurer. *Autool Version 2 Manual for COSTOC Authors.* Report, IIG Graz, 1987.

[ISO] ISO/IEC CD 10744. *Information Technology - Hypermedia/Time-based Structuring Language (HyTime).* Technical Report, International Organization for Standardization, Geneva, 1991.

[Lam] L. Lamport. LaTeX, *A Document Preparation System, User's Guide and Reference Manual.* Addison-Wesley Publishing Company, 1985.

[MMO] F. Makedon, H. Maurer, and Th. Ottmann. Computer learning: a step beyond the book. In B. Krause and ' Schreiner, editors, *HECTOR VOL I: New Ways in Education and Research*, Sprnger Verlag, Berlin, 1988.

[Ott] Th. Ottmann. Elektronische Kurse - Kritische Bilanz und Konsequenzen für die Werkzeugentwicklung. In P. Gorny, editor, *Informatik und Schule 1991*, pages 1-15, Springer-Verlag, Informatik-Fachbericht Nr. 292, Oldenburg, Germany, 1991.

[Tom] F. Tompa. A data model for flexible hypertext database systems. *ACM Transactions on Information Systems*, 7(1):85-100, 1989.

[Ukk] E. Ukkonen. Algorithms for approximate string matching. *Information and Control*, (64):100-118, 1985.

[WM] S. Wu and U. Manber. Fast text searching allowing errors. *Communications of the ACM*, 35(10):83-91, 1992.

Interactive Technologies for Language Learning

Philip Barker, Susan Giller, Stephen Richards, Ashok Banerji and Chris Emery
Interactive Systems Research Group,
School of Computing and Mathematics, University of Teesside, Cleveland, UK

Increasingly, as people become more internationally mobile there is a growing demand for tuition in modern 'spoken' languages such as French, German, Dutch and Italian. In order to cope with this demand language instructors are now turning to the use of computer-assisted language learning (CALL) techniques based upon the use of interactive computer-based workstations (Cameron, 1989; Chesters, 1990; Pollard and Yazdani, 1991). In the past the use of CALL techniques has been severely hampered because of the limitations imposed by the delivery stations and courseware which had to be used; these were mainly text-based and there was only very limited end-user interaction and participation. Currently available technology now enables markedly improved CALL environments to be developed based upon the use of high-quality pictures, sound input and output, and moving images. These multimedia resources can be organised in either a linear or non-linear (hypermedia) fashion depending upon the objectives to be realised (Barker, Emery, Fox and Giller, 1992).

Some time ago we embarked upon an exploration of the potential of reactive pictures and multilingual audio resources as a means of augmenting the teaching of foreign languages using CALL methods (Barker, 1990). This work was based upon an electronic book metaphor (Barker, 1991). The project, which was called SPBAN (an acronym for Static Picture Books with Audio Narration), used high-quality static pictures and audio recordings organised in the form of a highly adaptable electronic book that users could tailor to their own particular requirements (Giller, 1992).

Although the research described above only involved creating experimental prototypes, commercial products embedding these ideas are now starting to become available at relatively low cost. One extremely good example of a product range that embeds the electronic book metaphor is the 'talking electronic books' series produced by the Discis corporation (Discis Knowledge Research, 1991). The electronic books that make up this product line are essentially multilingual, multimedia story books for children. They are published on compact disc read-only-memory (CD-ROM) for delivery (primarily) in an Apple Macintosh PC environment - although some of the books are also available for use with Commodore's Dynamic Total Vision (CDTV) system. Fundamental to the success of these books is their use of high-quality reactive pictures and text and, of course, their incorporation of multilingual digital sound.

Undoubtedly, within a learning and training environment the use of interactive multimedia and hypermedia techniques for CALL can lead to substantial improvements in the quality of linguistic experience that can be achieved (Fox et al, 1990). Therefore, in the remainder of this paper we discuss how digital multimedia optical storage technology can be employed in various ways to develop and distribute CALL resources and, in so doing, improve the quality of the training experience to which students are exposed. The paper describes two ongoing research and development projects which involve the teaching of French; some possible future directions of development for CALL are also outlined.

THE CALSA PROJECT

Early in 1989 the Learning Technology Unit of the United Kingdom's Training Agency (as it was then called) issued a call for research and development proposals relating to the application of leading edge technology to learning and training. In response to this, the University of Teesside put forward a project proposal which involved exploring the potential of interactive multimedia and hypermedia CALL techniques to the problems of computer assisted (foreign) language study and acquisition. The major objective of the project (which became known as the CALSA project) was the creation of a language learning environment capable of providing for the needs of a variety of potential students both inside and outside higher education (Ingraham and Emery, 1991).

One of the long-term objectives of the CALSA project was to develop CALL resources to support the study of each of the European languages by 'native' English speaking students. However, because of the timescale of the present funding emphasis is currently being given to just one language - French. Twenty modules are being produced; each one contains four one-hour interactive multimedia units and an optional small group/tutor-based session. Some non-computer-based audio, video and paper-based materials may also be available. The pedagogic resources that are being developed are all based upon the 'communicative competence' model of language acquisition; that is, instilling in students the basic skills needed to communicate with and understand a foreign language within the functional context of professional and industrial life.

The overall structure of the courseware system to facilitate the interactive practice sessions referred to above is illustrated schematically in figure 1 (Ingraham and Emery, 1991). The system is based upon a 'book' metaphor similar to that described by Barker (1991). However, this model greatly extends and enhances the basic book metaphor in many interesting and important ways. It does this through the introduction of different types of book to support different language functions - for example, lesson books, a grammar book, a pronunciation book, a talking dictionary, and so on. The courseware itself is written in an authoring environment called ToolBook for delivery within an IBM PC environment which is capable of supporting Microsoft's Windows 3 graphical user interface (Fox, 1990).

An important aspect of the workstations that are used to develop and deliver the CALL resources referred to above is their speech handling capabilities. High-quality digital multilingual sound can be delivered to students who can subsequently use the speech input and output facilities to practise their own pronunciation of phrases and sentences. In the workstations that will ultimately make up the interactive learning environments that students use the digital sound (along with the other courseware resources) will all be stored on CD-ROM.

THE INTERACTIVE LANGUAGE LEARNING PROJECT

Running in parallel with the CALSA project is a second interactive language learning project (ILLP) which is also funded by the Department of Environment's Learning Technology Unit. This project (which complements the CALSA Project) commenced in late 1991; its intent is to investigate and evaluate the potential of various digital optical storage technologies (based upon CD-ROM) for the dissemination and delivery of CALL resources. This project is being run in conjunction with the UK's National Council for Educational Technology (NCET).

CD-ROM offers a high capacity integrated storage medium for the storage and dissemination of learning resources for use in many learning situations. This technology is particularly useful for the support of distance learning, open learning, and use in the home. Conventional CD-ROM allows the storage of significant quantities of text, multilingual sound, high quality pictures and cartoon

ERIC

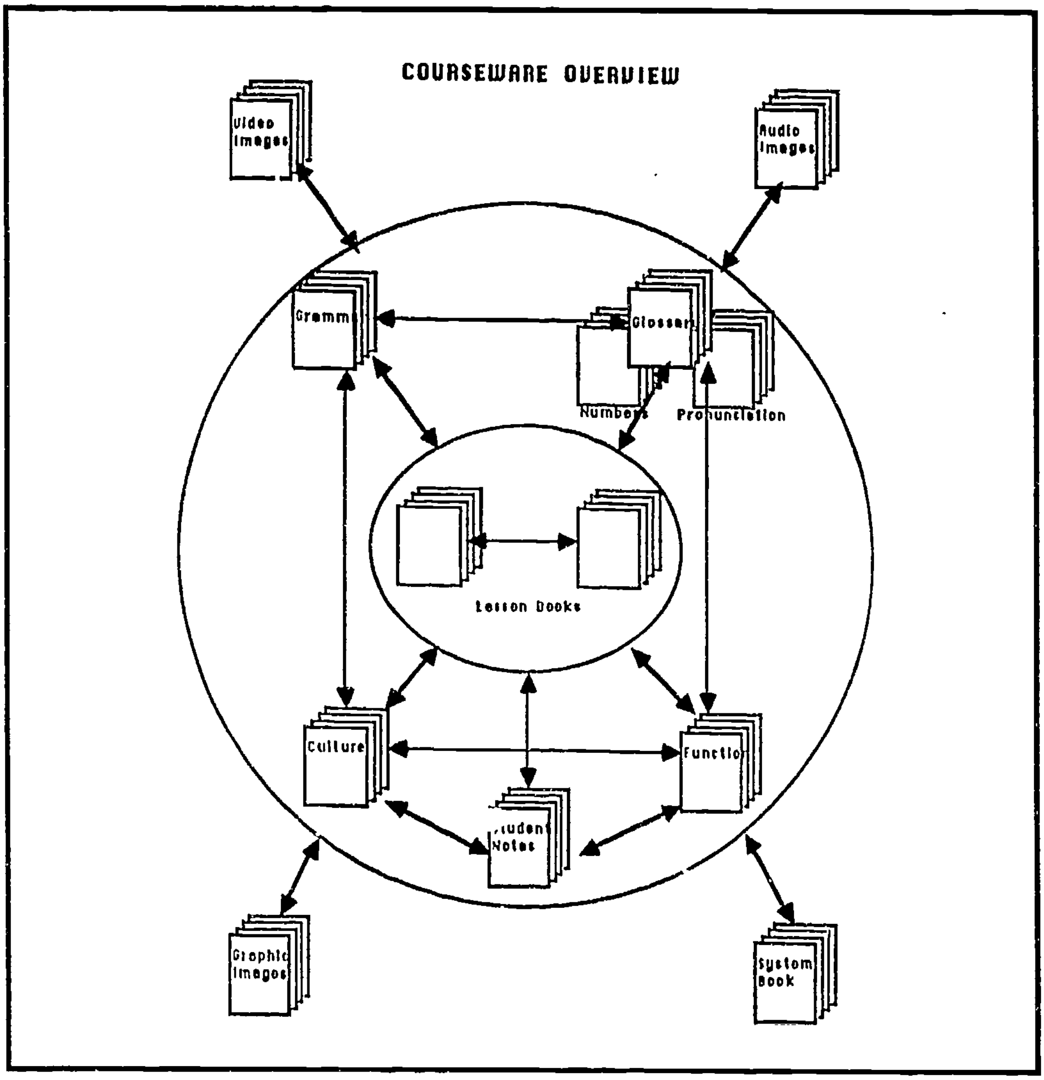


Figure 1. Application of the electronic book metaphor to interactive language learning

animation. Extensions to the basic CD-ROM facility enable TV quality motion video pictures to be utilised. This project is intended to explore the ways in which these digital resources can be used for the creation of interactive learning environments to facilitate foreign language acquisition and study.

Currently, there are five basic approaches to using digital optical storage media on a low-cost PC: (1) applications of conventional CD-ROM; (2) use of CD-ROM extended architecture (CD-ROM XA); (3) use of Commodore's Dynamic Total Vision system (CDTV); (4) applications of CD-I (compact disc - interactive); and (5) use of digital video interactive (DVI). At the onset of the project we thought that the latter two technologies were not sufficiently widespread within schools and colleges to warrant their use within this project. We therefore chose to explore the potential of conventional CD-ROM, CDTV and CD-ROM XA because equipment to support these technologies could be purchased commercially at relatively low cost.

The major advantages of using CD-ROM technology for instructional systems stem from the potential portability that this medium offers and the low-cost development and delivery stations that can be produced - based on existing computer workstations (CD-ROM and CD-ROM XA) or equipmen: that is available in the consumer marketplace (CDTV).

Bearing in mind what was said above, the major objectives of this project were: (1) to create three demonstrator systems that could be used to compare the relative merits of CD-ROM, CDTV and CD-ROM XA as media for the storage and distribution of interactive instructional material; and (2) to evaluate the potential utility of these technologies for the delivery of instructional materials related to the teaching of foreign languages - particular emphasis being given to French.

Unfortunately, after the project commenced and got underway we realised that the amount of development effort that was needed to produce three evaluation platforms was far in excess of what was realistically available to us. The objectives of the project were therefore re-written so as to require the creation of just two delivery platforms. One of these (as before) would use basic CD-ROM to deliver interactive text, digital sound and high quality static pictures. The other platform would also deliver these materials and, in addition, it would also include a number of motion video clips that were to be delivered using Intel's ActionMedia DVI boards.

In order to realise the newly defined objectives we prepared a 30 minute instructional unit based upon a series of situation scenarios (Pollard and Yazdani, 1991). The courseware for this experimental lesson was developed using Authorware Professional (running within a Windows 3.1 environment) for delivery using IBM PC compatible machines that utilised Intel 80486 processors. When the courseware was complete it was published on CD-ROM - fifty copies of each of two master discs were produced. These were then evaluated in a number of schools and colleges.

The evaluation phase of this project had two main objectives. First of all we wanted to try and measure skill acquisition across the four generic language skills: speaking; reading; listening and writing. It was intended that learner skill acquisition using the two platforms should be compared to that of a control group which received normal teaching with conventional materials. Data gathering on this was undertaken by a post-test constructed to examine mastery of the courseware content. The second objective was to compare the platforms against each other, and measurements here were to include: usability of the system; attitudes and motivation; individual or group work preferences, learning styles and feedback on the role of the tutor. The basic techniques used for these measurements included: observations; interviews and questionnaires - that were refined as a result of a pilot study. The second evaluation objective meant that in the trialling, groups of learners had to have access to each hardware platform after completing the post-test following their work on one of the systems (basic CD-ROM or DVI. As resources of people and time were strictly limited on the project, trialling the systems took place in only three centres. These included a polytechnic or college of higher education with post-18 students, a further education college (or a sixth form college) with students in the 16-19 age range, and a group of adults on a company language training programme. It was assumed that learner entry competence approximated to levels 6 or 7 of the National Curriculum Key Stage 4 in French (that is, people who had obtained the equivalent of a grade C at GCSE). The material was therefore not aimed at ab initio learners, but at students wishing to continue with their language studies either for interest as part of a core entitlement, or for vocational/academic purposes.

Unfortunately, the timetable of the project only allowed for the trialling centres to have the systems for a couple of weeks each, during which time the basic unit of work had to be covered, observations made, questionnaires filled in, the post-test completed and interviews conducted. Some of the key questions to which we were seeking answers were: does the courseware presented on these systems allow greater skill acquisition faster than the materials presented in the normal way; was there more of a pay off in one of the language skills than in the others; do the technologies and the material facilitate group interaction or individual study; what are the attractive features of the media; what are the preferred learning styles; was there a difference in usability between the three platforms; was there an increase in motivation; and what was the role of the tutor using interactive technologies with the groups of learners?

FUTURE DIRECTIONS

Situation scenarios of the type used in the ILLP courseware described above offer a powerful mechanism for implementing the communicative competence model for langauge acquisition and study. However, as with all 'conventional' simulation environments the effectiveness of learning is always influenced to some degree or another by the level of realism that can be created. For this reason we are now looking for ways and means of substantially improving the degree of realism that we can create within our situation scenarios for language learning. With this end in view we have recently been exploring the potential of artificial and virtual reality (cyberspace) systems (Krueger, 1991; Helsell and Paris Roth, 1991).

Cyberspace systems provide a mechanism by which instructional designers can create 'artificial' (computer-generated), time variant, three-dimensional worlds. Within these worlds learners become active participants interacting (in real-time) with other objects by means of speech, touch and gestures. Ideally, from the point of view of language teaching, the quality of experience encountered in a cyberspace environment will be indistinguishable from those experienced in real environments such as a cafe, a pub, a library or a railway station. The only difference will be that the cyberspace environments will be computer-generated from blue-prints that have been laid down by instructional designers. Such systems, although costly to build at present, will have many important roles to play in interactive language learning in the future.

CONCLUSION

The design of computer-based interactive learning environments to support foreign language acquisition and study places considerable demands on designers of instructional systems. Indeed, until recently the design, production and delivery of CALL resources has been severely hampered by many technical barriers. However, the advent of low-cost interactive workstations which incorporate digital optical storage facilities now makes available many new approaches to language instruction using CALL techniques. This paper has described two ongoing projects in which we are currently involved. The intent of each project, at their outset, was to investigate ways of improving the quality of the CALL experiences to which students could be exposed through the use of interactive multimedia and hypermedia techniques.

Obviously, the use of these techniques within CALL environments has led to a substantial improvement in the quality of linguistic experience that can be achieved within automated language instruction systems. However, there are still many problems to be solved if we wish to improve the quality of the simulated environments to which students are exposed. In this context cyberspace technology has much to offer

ACKNOWLEDGEMENTS

Much of the work described in this paper has been funded by the Training, Enterprise and Education Directorate (TEED) of the United Kingdom's Department of Employment. The authors gratefully acknowledge this financial support and also the enthusiastic encouragement of all their colleagues at the University of Teesside and NCET (particularly, Jeff Morgan and Leslie Glover). We are also indebted to Alan Clarke and Keith May of TEED for their sound advice and enthusiasm.

REFERENCES

Barker, P.G., (1990). Human-Computer Interface Design for Electronic Books: a Case Study in Foreign Language Learning, Proceedings of CD-ROM Europe '90, Agestream Ltd, London, May 1990.

Barker, P.G., (1991). Interactive Electronic Books, Interactive Multimedia, 2(1), 11-28.

Barker, P.G., Emery, C., Fox, P. and Giller, S., (1992). Multimedia Support for CALL, Proceedings of the Ninth International Conference on Technology and Education, Paris, 16th-20th March, 1992.

Cameron, K., (ed), (1989). Computer Assisted Language Learning, Intellect Books, Oxford, UK.

Chesters, G., (1990). ReCALL Software Guide (Issue 2), The CTI Centre for Modern Languages, The University of Hull, Hull, HU6 7RX, UK.

Discis Knowledge Research Inc, (1991), 45 Sheppard Avenue East, Suite 410, Toronto, Ontario, Canada, GU1 3UW.

Fox, J., (et alia), (1990). Educational Technology in Modern Language Learning, Learning Technology Unit, Training, Enterprise and Education Directorate, Department of Employment, Moorfoot, Sheffield, S1 4PQ.

Fox, P., (1991). Multimedia Support for CALL, Outline MPhil Proposal, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Giller, S., (1992). Design Guidelines for Electronic Book Production, MPhil Dissertation, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Helsel, S.K. and Paris Roth, J., (1991). Virtual Reality: Theory, Practice and Promise, Meckler, Westport, CT 06880, USA.

Ingraham, B. and Emery, C., (1991). 'France InterActive': a Hypermedia Approach to Language Training, Educational and Training Technology International, 28(4), 321-333.

Krueger, M.W., (1991), Artificial Reality II, Addison-Wesley, Reading, MA, USA.

Pollard, D. and Yazdani, M., (1991). A Multilingual Multimedia Restaurant Scenario, Interactive Multimedia, 2(4), 43-51.

The Evaluation of Multimedia Courseware

Philip Barker, Susan Giller, Stephen Richards and Terry King
Interactive Systems Research Group,
School of Computing and Mathematics, University of Teesside, Cleveland, UK

The successful utilisation of interactive computer-based technologies within education depends critically upon the design of appropriate learning environments and the establishment of the infra-structures that are necessary to facilitate their use. An interactive learning environment is one in which either individual or group learning processes are strongly influenced by the nature of the real-time feedback that is obtained from the learning environment itself (Barker, 1990; 1993). Such environments can vary quite enormously in complexity - from simple drill and practice (on a microcomputer) through computer supported cooperative learning (using computer networks) to virtual reality environments (based upon the use of sophisticated cyberspace decks).

Most conventional interactive multimedia learning environments depend upon two basic types of resource for their effective and efficient utilisation. First, a suitably designed workstation facility which allows students to interact (both with each other and with the embedded learning resources). Second, the availability of appropriate multimedia courseware products that are able to utilise the workstation resources in the most appropriate way. Each of these issues is further discussed below.

Multimedia Workstation Environments

The workstation environment needed to support the use of multimedia courseware requires six basic types of component: (1) a low-cost, high-speed computational ability; (2) a high-resolution screen to facilitate the display of both static and dynamic pictures and the use of multi-font text of various types; (3) appropriate interaction devices (such as a mouse, touch-sensitive screen, tracker-ball or roller controller) to enable users to perform on-screen pointing operations; (4) large-capacity, fast-access storage devices such as those based upon optical disc; (5) facilities for acquiring, storing, processing and generating sound; and (6) suitable connections to local-area and wide-area communications networks. Within Europe we are attempting (through the European Community's DELTA project) to identify the most effective way of using these components to produce a multimedia interactive computer environment for educational purposes. The architecture that is being developed is called the 'common training architecture' (CTA).

Undoubtedly, the majority of computer workstations available today fulfil most of the requirements listed above. Indeed, as we move into the mid-90s we are seeing the development of truly multimedia computers. That is, computers which can support the use of text, static pictures, moving pictures of various sorts, sound - and a variety of different styles of human-computer interaction. Such computers therefore have the ability to acquire, store, process, distribute and deliver various types of interactive multimedia information. From the point of view of courseware design this is extremely important since it provides designers with the ability to mix such information in a multitude of different ways and to use those 'media mixes' that are most appropriate to the needs of particular learning applications.

Multimedia Courseware

Much of the early courseware used for computer-assisted learning (CAL) and computer-based training (CBT) depended primarily upon the use of linearly organised screen-based text for the delivery of instructional material (Dean and Whitlock, 1992). However, as the ability to incorporate sound effects and graphics became available, so courseware took on a more multimedia nature (Barker, 1989). In this paper we use the term multimedia to refer to the ability to combine (within a given application) multiple channels of communication within a single distribution or delivery medium - such as magnetic or optical disc. Multimedia courseware must therefore employ two or more communication channels (either together or in sequence) and use these in appropriate ways in order to achieve its pedagogic objectives. As we have suggested above, the basic channels with which we are concerned may be of a textual, sonic, graphic or haptic nature.

Of course, in some learning or training situations it might be necessary to use several different and distinct delivery media (such as a paper workbook and a computer screen) in either a simultaneous or a sequential way in order to achieve particular types of learning objective. We use the term 'polymedia courseware' to describe instructional resources of this type in which several different kinds of information delivery medium are used within a given learning or training application.

As well as giving interactive courseware a multimedia character, the workstation developments described in the previous section have also made it possible to interlink instructional materials in a variety of sophisticated ways. This interlinking enables it to be accessed and delivered using many different non-linear approaches. Because of the various advantages that are claimed for the use of hypertext and hypermedia, courseware for many learning applications is increasingly employing hypermedia techniques.

Naturally, the use of multiple channels of communication (on single or multiple media and using either a linear or a non-linear organisation) implies the need for considerable care with respect to the way in which these channels are employed. For example, it is important to consider such issues as synchronicity, continuity, cohesiveness, reinforcement and appropriateness. As we shall discuss later in this paper, design guidelines for using multimedia and hypermedia methods within courseware are therefore an important requirement. In order to obtain these guidelines effective methods of evaluating multimedia courseware must be devised and applied in both a formative and summative fashion. In this paper we describe some of the approaches that we have been using to evaluate and assess multimedia courseware resources that have been produced by the various projects upon which we are working.

COURSEWARE EVALUATION

The evaluation of interactive multimedia courseware can involve a range of different 'dimensions of interest'. Typical dimensions along which courseware may be evaluated include: learning effectiveness; cost effectiveness; interactivity; interaction styles; synchronicity; continuity; cohesiveness; interface quality; ease-of-use; availability; ease-of-access; the quality of learning experiences involved; level of embedded intelligence; the effectiveness of media utilisation; the nature and level of ancillary support needed to use the product; the complexity of the delivery environment needed to support the courseware; and so on. Because of the multi-dimensional nature of courseware evaluation it is often appropriate to 'weight' individual dimensions according to the perceived needs of the evaluation task.

Because of the importance of courseware evaluation many different assessment tools and methods have been described in the literature. Within these, various approaches are used as instruments of evaluation - such as checklists, questionnaires, structured interview schemes, pre-test and post-test strategies, and so on. Some typical examples of evaluation procedures include the MEDA Tool (Machell and Saunders, 1991), those documented by Bonekamp (1991) in the ECOSET project (which is a part of the European Community's DELTA initiative), the methodology of Ravden and Johnson (1989) and the '3G' approach described by Collis (1991) for the evaluation of electronic books. In the work that we have undertaken we have employed a variety of approaches based upon the questionnaire method and pre-testing/post-testing experiments.

In the remainder of this section we describe four evaluation projects that we have been conducting - the intent of which is to assess the quality of various types of interactive multimedia courseware (along particular dimensions of interest). The various projects investigate: conventional electronic books; hypermedia electronic books; resources for interactive language learning; and learning design for interactive compact disc.

Electronic Books

Electronic Books offer an important mechanism for the storage and delivery of substantial volumes of multimedia information - particularly, if they are published on compact disc read-only-memory (CD-ROM). Of course, they can also be used as an adjunct to conventional CAL and CBT and so can embed various types of multimedia courseware. For some time we have been interested in using electronic books as an aid to support various approaches to open and distance learning by means of 'telemedia books' and electronic performance support systems (Giller, 1992; Barker and Banerji, 1993).

Our 'Electronic Book Project' (which has now been running for over two years) was set up in order to explore, investigate and evaluate various types of electronic book with a view to formulating a set of design guidelines which authors and producers could use in order to produce successful electronic book productions. Two basic research strands have been pursued. First, the use of various controlled evaluations of a number of existing commercially available electronic books. Second, the design, production and controlled evaluation of three in-house electronic book productions. The evaluation metric and methodology that was used in each of the two types of evaluative study was identical.

The majority of commercial electronic books that were evaluated were published on CD-ROM although some were distributed on magnetic disc (as native systems or in compressed format). The publications that were examined included: Hypertext Hands-On; the Grolier Encyclopedia; Compton's Multimedia Encyclopedia; the Oxford Textbook of Medicine on CD-ROM; The Complete Works of Shakespeare; Sherlock Holmes on Disc; World Atlas; the Nimbus Music Catalogue; Multimedia Mammals; and Multimedia Birds of America. The three in-house productions involved the creation of: a static picture book with audio narrations (for language teaching); an electronic book for early learners (based upon the use of hypermedia); and an electronic textbook on screen design for CBT.

The evaluation studies produced many interesting results from which were derived some valuable design guidelines for electronic book production.

Hypermedia Electronic Books

Traditionally, the majority of electronic books (similar to most of those described above) provide linear access to stored information. Of course, such books can also usually support both direct and branching access through the provision of appropriate search facilities. In contrast to the linear approach, hypermedia electronic books provide access to multimedia resources that are organised in the form of network information structures. Links embedded within information items allow users to follow pathways through the linked material. In this way, they are able to follow up ideas or items of interest in much the same way as one might follow a train of thought.

Our work with hypermedia books involves assessing the impact of employing the hypermedia paradigm within multimedia electronic book environments (Richards, 1992; Barker, 1992). Parallel electronic book applications employing hypermedia and traditional delivery strategies have been developed. These systems have involved embedding hotspots in both textual and graphical information. Up to twenty hotspots are embedded in each of the information nodes within the information space through which users can navigate. Additional navigation facilities consistent with the book metaphor are also provided. These include: page turning; a table of contents; use of an index; the incorporation of page numbers; and so on.

Evaluations are being conducted which involve applying suitably designed assessment metrics in order to quantify and compare users' responses when undertaking various types of information access task. We are particularly interested in the impact that hypermedia techniques might have when students use our electronic books as either reading or reference resources. Both qualitative views and quantitative performance data are being collected.

Subjects undertaking the evaluation experiments (that are designed to assess their performance on reading tasks) are required to take both a pre-test and a post-test. These allow an assessment of the subjects' initial knowledge to be made and compared with the levels of knowledge displayed in the post-test. A comparison between the performance of subjects who use the hypermedia system and the performance of those who use the traditional electronic book system is now being made.

In order to assess the usefulness of hypermedia techniques in the context of using electronic reference resources another set of experiments was designed. These are based upon giving subjects a number of information retrieval tasks involving the use of the parallel information delivery environments for looking up specific items of information. Both the time taken and the accuracy of the retrieved information are being monitored and assessed.

In both the reading and reference evaluations (outlined above) subjects are also being given a qualitative evaluation questionnaire in which they can express their views about the experiments in which they have participated. The questionnaires are designed in a way which will enable us to assess users' reactions towards the traditional and hypermedia delivery strategies.

The results of the evaluations are providing a valuable profile of the potential impact of hypermedia information access and the types of task to which it might be most suited within the context of electronic book design.

Interactive Language Learning

Over the last two decades computer-assisted language learning (CALL) has had considerable success with respect to many aspects of language teaching (Ingraham and Emery, 1991). Obviously, the teaching of languages is a participative, multimedia activity requiring (at the very least) the use of sound and text. We anticipate therefore that the quality of CALL experiences available for students can be substantially improved as facilities for handling sound and pictures become available (Fox, 1992). Important techniques that need to be considered include: the use of CD-ROM for storing and delivering courseware; the application of low-cost sound-boards; the appropriate use of reactive static graphical images; and the inclusion of digital motion video - in situations where it is necessary and/or appropriate.

Like many other academic organisations we have been extremely interested in the use of CALL methods for the teaching of foreign languages. Indeed, we have recently become involved in two complementary research and development projects called 'CALSA' and 'ILLP'. CALSA is an acronym for 'Computer Aided Language Study and Acquisition'; it is a project that uses the latest developments in computing and information technologies in order to develop substantial volumes of CALL resources for the teaching of European languages (Ingraham and Emery, 1991). Naturally, CD-ROM and interactive video are used extensively in this project because of the very high quality of sound and images that can be stored and retrieved from these media.

Our 'Interactive Language Learning Project' (ILLP) is designed to assess the relative merits of some of the currently available digital optical storage media for use in interactive CALL delivery platforms (Barker et al, 1992). A range of possibilities exist: basic CD-ROM; DVI; CDTV; CD-I; CD-IV; CD-ROM XA; CD+G; and so on. In our ILLP evaluation study we have selected the first two of these for assessment (basic CD-ROM and DVI). A short, 'standard' piece of language instruction was designed for the evaluation experiments which involved native English speakers; French was selected as the target language. The lesson is of a multimedia nature - relying on the use of text, sound and pictures - and was produced for delivery using each of the selected delivery technologies (each technology being used to its maximum capability). Two CD-ROM discs were fabricated. These have been trialled and the lessons evaluated using both questionnaire methods and pre-testing and post-testing experiments. The latter were run in conjunction with a control group that was taught using conventional approaches to language learning.

Integrating Learning Design in Interactive CD

Through its DELTA programme the European Commission is currently funding twenty-two major research and development projects. These projects cover a range of technical and educational issues relating to the application of advanced information processing technologies to the problems of creating effective and efficient distance and open learning environments based upon the use of interactive computer systems. One of these projects (in which we are involved) is intended to examine the Integration of Learning Design in Interactive Compact Disc (ILDIC).

The two major objectives of the ILDIC project are: (1) to develop and evaluate models of learning on interactive multimedia in order to build prototypes in hypermedia and 'bridge format' for compact disc (CD); and (2) to produce standards for the design and authoring of learning applications on interactive CD that meet the specifications laid down by the common training architecture that has been proposed within other DELTA projects.

The ILDIC project (which involves partners in the UK, Greece and Denmark) contains a number of basic research strands (evaluation, synthesis, design, and implementation). The strand in which we have been working involves the evaluation of a range of currently available interactive courseware products (King and Barker, 1992). The particular products that are of interest to us are multimedia systems that utilise compact disc as a dissemination and delivery medium. In the work that we have undertaken we have employed an enhanced version of a 'questionnaire' method previously used by Giller for the evaluation of electronic books that have been published on compact disc (see above).

Using Giller's basic 'electronic book' approach, we have designed and tested an appropriate evaluation metric and methodology for the dimensions of interest which are of primary concern to us. Essentially, this summative evaluation procedure is based on the use of 'prompt questions' which an evaluator administers alone, to individuals or to groups of users who have been invited to assess particular courseware products. The results obtained from individual product assessments are then collated and analysed with a view to identifying the outstanding features of particular products and/or their major limitations.

The evaluation methodology has been applied to 35 commercially available products that cover a range of delivery media such as basic CD-ROM, CDTV, DVI and CD-I. Taken together these

ERIC

products represent an extremely wide spectrum of approaches to the provision of interactive learning resources that involve a very rich repertoire of pedagogic methods. Typical learning methods involved in these products include: games; simulations; role playing; tutorial approaches; encyclopedia access and exploration; and conventional computer-based training.

The evaluations that have been conducted within the ILDIC project have provided us with a range of useful results which have enabled us to gain considerable insight into the nature of successful design for learning and training products that are based upon the use of interactive compact disc.

CONCLUSION

Interactive multimedia courseware can take a variety of different forms and can embed a wide range of pedagogic mechanisms. Consequently, courseware can be used to perform a multitude of educational functions ranging from the generation of awareness through record keeping to skill development and enhancement. Because the intent of courseware is to teach and develop skills it is important that suitable evaluation methods should exist in order to assess to what extent, if at all, these objectives are being realised. As we have established in this paper, evaluation is a multi-faceted activity. It can be undertaken with respect to many different dimensions. In order to keep the evaluation task manageable, those involved in the design and implementation of evaluation experiments may have to identify the most important dimensions of concern and ignore those which are of less importance. Of course, in order to perform cross-product comparisons some form of weighting algorithm may need to be imposed in order to make the results meaningful.

The pedagogic dimensions of courseware evaluation are of paramount importance because unless interactive multimedia instructional software has some educational value there is little point in using it. However, as well as being pedagogically effective, courseware must also be relatively easy to use and, whenever possible, it should create an enjoyable and motivating learning experience for its users. These latter requirements can usually be achieved through the incorporation of appropriate human factors considerations when the courseware is designed.

In the work that we have undertaken (as outlined in the four case studies presented in this paper) we have attempted to develop a multi-faceted courseware evaluation methodology that will enable us to assess the quality of both: (1) the interactive multimedia courseware that we have been producing; and (2) that which is available commercially. The results that we have achieved to date indicate that the evaluation methodology is sound and gives a reasonable indication of courseware quality with respect to the particular dimensions of interest that are of importance within any given situation.

ACKNOWLEDGEMENTS

The authors are indebted to all those people and organisations who have contributed technical help and financial aid to support the work described in this paper. We would particularly like to thank: Cambridge Training and Development Ltd (support for Philip Barker and Terry King); Dean Associates and the UK's Science and Engineering Research Council (support for Stephen Richards); and the UK's Department of Employment (through its Learning Technology Unit) for support for Susan Giller.

ERIC

REFERENCES

Barker, P.G., (1989). Multimedia Computer-Assisted Learning, Kogan Page, London.

Barker, P.G., (1990). Designing Interactive Learning Systems, Educational and Training Technology International, 27(2), 125-145.

Barker, P.G., (1992). Hypermedia Electronic Books, paper presented at the 7th Canadian Symposium on Instructional Technology, Montreal, May 1992.

Barker, P.G., (1993). Interactive Learning and Technologies to Support It, SCED Working Paper, Interactive Systems Research Group, School of Computing and Mathematics, University of Teesside, Cleveland, UK.

Barker, P.G. and Manji, K.A., (1993). Multimedia Environments for EPSS Development, 423-430 in Proceedings of the Indian Computing Congress (ICC '92) - Innovative Applications in Computing, edited by E. Balagrurusamy and B. Sushila, Tata Mc-Graw-Hill Publishing Company, New Delhi, India.

Barker, P.G., Fox, P., Emery, C., Giller, S., Morgan, J. and Glover, L., (1992). Interactive Technologies for Language Learning, paper presented at CD-ROM Europe 92, Wembley Conference Centre, London, October 1992.

Bonekamp, L.W.F., (1991). Summative and Formative Evaluation Activities Regarding Courseware in Europe: Results of the ECOSET Project, Educational Computing Consortium BV, PO Box 217, 7500 AE Enschede, Holland.

Collis, B., (1991). The Evaluation of Electronic Books, Educational and Training Technology International, 28(4), 355-363.

Dean, C. and Whitlock, Q., (1992). A Handbook of Computer-Based Training, Third Edition, Kogan Page, London.

Fox, P., (1992). Multimedia Support for CALL, Draft MPhil Thesis, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Giller, S., (1992). Design Guidelines for Electronic Book Production, MPhil Thesis, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Ingraham, B. and Emery, C., (1991). 'France InterActive': a Hypermedia Approach to Language Teaching, Educational and Training Technology International, 28(4), 321-333.

King, T. and Barker, P.G., (1992). DELTA ILDIC Project, Investigation of Learning Design in Existing Multimedia Learning Materials - Report on Evaluation and User Trials, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Machell, J. and Saunders, M., (1991). MEDA: an Evaluation Tool for Training Software, CSET, University of Lancaster, UK.

Ravden, S.J. and Johnson, G.I., (1989). Evaluating Usability of Human-Computer Interfaces: A Practical Method, Ellis Horwood, Chichester, UK.

Richards, S. (1992). Human-Computer Interfaces to Electronic Books, Draft PhD Thesis, Interactive Systems Research Group, University of Teesside, Cleveland, UK.

Audio Instruction in Multimedia Education: Is Textual Redundancy Important?

ANN E. BARRON
Instructional Technology Program
University of South Florida, Tampa, FL 33620, USA

DEBRA ATKINS
Florida State University, Tallahassee, FL 32312, USA

Introduction

Until recently, the majority of the computer-based training (CBT) programs concentrated on the visual presentation mode to the exclusion of the auditory mode because of hardware constraints. In fact, many textbook authors that purported to teach the design and development of CBT and multimedia failed to even mention the auditory channel and certainly did not focus on its potential.

The advent of the Macintosh computer with audio capability and availability of moderate cost, good quality digital audio cards for MS-DOS computers have enabled trainers and educators to realize the potential of random access audio for CBT and multimedia. Access to the audio segments is rapid, and the amount of information stored can be very large. Control of the audio files through computer software programs is precise and reliable.

Another major advantage of digital audio is its flexibility. New dialogue can be recorded, digitized, and inserted into an existing program easily, providing much greater program longevity. With just a few changes to the script and a few minutes with an audio editor, a new audio file can replace an old one, providing necessary revisions and updates to software.

There are, however, few guidelines for instructional designers to follow with incorporating audio into computer-based lessons. In the past, research studies investigated the potential increase in learning performance based on a multichannel delivery; however, few studies existed that correlated multichannel effectiveness and redundancy levels of computer-based programs. A previous study by the author indicated that there was no statistical difference in achievement (immediate recall) between a text-based tutorial program and a combination of text/audio program. There was, however, a significant difference in the delivery time required for the lessons (Barron & Kysilka, 1992). Based on these initial findings, a study was designed to further investigate possible relationships between various levels of audio and textual redundancy on student achievement, retention, and time.

Research Questions

The study presented in this proposal was designed to provide answers to the following questions:

- Does the amount of textual redundancy in audio-based multimedia produce a statistically significance increase in learning performance?
- Does the amount of textual redundancy in audio-based multimedia produce a statistically significance increase in retention?
- Does the instructional time required to complete an audio-based lesson vary in relation to textual redundancy?
- Does the amount of textual redundancy affect students' perceptions of the lesson?

Review of the Literature

Multichannel redundancy has been studied for many years, primarily using the television and videotape media. Little research, however, is available that investigates the effects of redundancy between text and audio on learners' achievement levels in computer-based training or multimedia instruction. An underlying assumption, when adding an additional channel of media to transmit a message, has often been that utilization of additional channels will effectively increase the amount of communication and the amount of learning. Reviews of research in this area, however, indicated that studies in multichannel communication resulted in mixed and contradictory findings (Jasper, 1991). Reasons cited for the controversy included poor sampling techniques, weak designs, lack of randomization, test channel bias, and variations in the definition of "redundancy."

The cue summation theory predicts that learning increases as the number of available cues or stimuli are increased (Severin, 1967). A proponent of this theory and multiple channel presentation, Hartman (1961) evaluated several studies with regard to simultaneous audio-print and print. Seven studies supported simultaneous presentation. Hartman concluded that "it is apparent that a simultaneous audio-print presentation is more effective than either audio or print alone when the information simultaneously presented is redundant" (p. 244). A more recent study by Nugent (1982) supported these findings.

Another theory that has been hypothesized (Travers, 1964) is based on an assumption by Broadbent (1958) that there is a single channel linking senses to the central nervous system. If this were the case, there would be little or no advantage to communication through two channels. Studies by Barton and Dwyer (1987) and Muraida & Spector (1992) found little or no difference in achievement levels based on multiple channel deliveries.

Due to the controversy of the research, there are few instructional design guidelines for the optimal relationship between text and audio in CBT and multimedia. For example, even if audio is chosen as a the primary delivery medium, to what degree should redundant text be incorporated for effective instruction? Designers must decide if text is essential, and, if so, whether the text should mirror the audio information or be condensed to bullets. Few of the studies found in the literature review had specifically evaluated redundancy factors associated with the integration of audio into computer-based programs.

Methods and Data Source

This study was conducted at the University of South Florida, Tampa, Florida, during the Summer semester 1992. The population consisted of students enrolled in EME 4402, Computers in Education, in the College of Education. The sample was constructed on the basis of 100 randomly assigned undergraduate students.

Basic instruction used in this study consisted of a computer-based lesson on Compact Disc-Read Only Memory (CD-ROM) technology (Barron, 1992). In order to provide three treatment programs necessary for this study, the CD-ROM Tutorial was developed in three alternate designs:

1. An audio-based version with total textual redundancy--the audio is word for word, the same as the instructional text on the screen.
2. An audio-based version in which the audio and graphics are the same as version #1, but the visual text of #1 is reduced to bulleted items, rather than full text.
3. An audio-based version in which the audio and graphics are the same as version #1 and #2, but there is no visual text.

ERIC

Subjects were randomly assigned to an experimental group. The three groups were pretested on the dependent variable--herein, achievement scores. There was no time limit to complete the lesson. Each treatment group was provided with an identical verbal introduction and computer-based overview of system usage. For delivery of all versions, headphones were worn by students to eliminate interference or overlap between treatments. Within the computer program, records were kept of the time each student started and the time he/she finished the tutorial.

Following the completion of the independent variable (one of the three audio- based programs), an achievement-based posttest was administered to all groups. Immediately following treatments, a post-experiment perception questionnaire was also conducted. In addition, a parallel achievement-based retention test was administered 3 weeks following the treatments.

The analyses incorporated one-way ANOVAs and a Repeated Measures MANOVA. The one-way ANOVAs were based on treatment groups (full text, partial text, or no text) as the predictor variable and time, pretest scores, posttest scores, and retention scores as the criterion variables. Tukey-b was used to analyze separate group differences when significant. The repeated measures MANOVA compared the changes of pretest scores, posttest scores, and retention scores over time. The level of significance for the analyses was set at .05.

Data Analysis

Achievement

Results of pretest, posttest, and retention tests indicated that all three treatments were instructionally very effective. Data for all three treatments increased significantly between the pretest and the posttest and then decreased significantly from the posttest to the retention test. All analyses exceeded the .001 level of significance (see Table 1).

TABLE 1
REPEATED MEASURES MANOVA FOR
PRETEST, POSTTEST, RETENTION TEST SCORES

TREATMENT	TEST	N	MEAN	SD	df	f
Full	Pretest	39	12.56	3.65	37	189.20*
Text	Posttest	39	24.54	2.88		
	Retention	39	17.92	6.99		
Partial	Pretest	30	13.07	2.73	28	158.04*
Text	Posttest	30	24.47	2.98		
	Retention	30	18.86	4.89		
No	Pretest	32	12.81	2.60	30	174.66*
Text	Posttest	32	23.81	3.44		
	Retention	32	18.44	5.81		

* $p<.001$

53

In order to ascertain that the three treatment groups were equal at the beginning of the experiment, a one-way analysis of variance (ANOVA) was performed on the pretest means. Table 2 contains a summary of the analysis between the three treatment groups.

TABLE 2
SUMMARY TABLE OF ANALYSIS OF
VARIANCE ON PRETEST BY TREATMENT

SOURCE	SUM OF SQUARES	df	MEAN SQUARE	F-RATIO	p
Between	4.302	2	2.1511	.2261	.798
Within	932.331	98	9.5136		
Total	936.634	100			

The results of the ANOVA on the pretest means indicates that the three groups did not differ significantly at the beginning of the experiment. ANOVAs were conducted on the posttest and retention test measures also (see Table 3 and 4).

TABLE 3
SUMMARY TABLE OF ANALYSIS OF
VARIANCE ON POSTTEST BY TREATMENT

SOURCE	SUM OF SQUARES	df	MEAN SQUARE	F-RATIO	p
Between	10.6393	2	5.3196	.5546	.5761
Within	940.0340	98	9.5922		
Total	950.6733	100			

TABLE 4
SUMMARY TABLE OF ANALYSIS OF
VARIANCE ON RETENTION TEST BY TREATMENT

SOURCE	SUM OF SQUARES	df	MEAN SQUARE	F-RATIO	p
Between	15.3346	2	7.6673	.2087	.8120
Within	3600.111	98	36.736		
Total	3615.446	100			

The results of the two tests indicate that the mean scores do not differ significantly between groups on either the posttest or retention test due to the treatment.

Time

The question pertaining to the effect of treatment on time was tested using a one-way analysis of variance with three treatments. The criterion variable used to assess completion time was the number of seconds each student required to complete all portions of the computer tutorial. Variables were written into the computer program to track the amount of time from objective screen to the exit screen. Students were not allowed to exit until all segments were complete.

TABLE 5
MEANS AND STANDARD DEVIATIONS FOR TIME (SECONDS)

	$\bar{X}$	SD	N
FULL	1220.00	208.78	39.00
PARTIAL	1224.10	222.36	30.00
NONE	1227.88	193.03	32.00

The results of the ANOVA measured by the completion times indicated mean scores did not differ significantly between treatment groups due to the amount of text in the delivery medium. Table 5 contains means and standard deviations of the completion times for each treatment, and Table 6 summarizes the one-way analysis of variance.

TABLE 6
SUMMARY TABLE FOR ANOVA ON TIME BY TREATMENT

SOURCE	SUM OF SQUARES	df	MEAN SQUARE	F-RATIO	p
Between	311.1267	2	155.5634	.0036	.9964
Within	4245404.2	98	43320.451		
Total	4245715.3	100			

Perception

A 20-item perception questionnaire was included as a post-experiment instrument. This was administered immediately after the posttest. Table 7 includes the frequency per treatment for each item included on the questionnaire.

TABLE 7
FREQUENCY OF PERCEPTION RESPONSES (AGREE/DISAGREE)

QUESTION	FULL		PARTIAL		NONE	
	A	D	A	D	A	D
I felt I could work at my own pace.	38	0	28	0	30	2
Hardware made it difficult to concentrate.	1	36	2	26	1	31
I enjoyed using the program.	36	2	27	1	30	1
I had trouble using the mouse.	7	31	1	27	5	27
It gave clear explanations of the material.	38	0	28	0	31	1
I enjoyed being able to hear the program.	37	1	26	2	31	1
I thought the audio was too slow.	12	26	7	21	2	30
The program made good use of examples.	38	0	27	1	30	2
The directions were easy to follow.	38	0	28	0	32	0
I would rather have had a lecture..	2	36	0	28	0	32
I thought the program was too long.	2	36	1	27	3	29
I'd encourage friends to take this lesson .	35	3	25	3	29	3
The Macintosh computer was easy to use.	38	0	27	1	31	1
The earphones were uncomfortable.	12	25	10	18	11	21
The audio was easy to understand.	37	1	27	1	30	2
I would prefer a program with no sound.	2	35	2	26	4	28
I do not like to read.	3	35	6	22	2	30
The program was boring.	1	36	0	28	2	30
I'd like to take another program like this.	35	3	28	0	27	5
Computer programs make me nervous.	8	29	6	22	7	27

Based on this questionnaire, the overall reaction to the CD-ROM tutorial program was extremely positive. For example, 86% "enjoyed using the program," 89% "would like to take another program like this," and only 2% "would rather have had a lecture in a classroom."

Five questions on the survey focused on the audio aspect of the programs. Ninety-seven percent of the students responded that "the audio was easy to understand," and 95% "enjoyed being able to hear the program." In addition, only 10% agreed that "the audio was too slow," and 13% "would prefer a program with no sound."

Chi-square analyses were conducted to determine if any student perceptions were significant in relationship to the three treatments. No tests of independence were significant.

Discussion

The audio-based computer tutorial designed to teach subjects about CD-ROM technology was quite successful. Pretest means indicated little prior knowledge and posttest means were significantly higher. Results further indicated that computer-based training with or without audio and textual redundancy was equally effective. None of the treatment groups demonstrated superior gains over the other groups. This lack of achievement differential indicates that the reduction of text in the partial-text/full-audio and no-text versions did not adversely affect achievement gains.

In many cases, the decision to incorporate audio into computer-based training is not driven by cost alone, but rather by screen real estate. In other words, in many instances, the best method of incorporating instruction with a complex graphic (such as a Navy sonar display) is to implement audio. The question has been raised as to whether this decrease in visual text would adversely affect learning. As indicted in this particular study, the impact is not significant.

There are several possible explanations as to why the level of achievement was unaffected by the amount of textual redundancy. First of all, this finding could provide support to Broadbent's theory that redundant multiple inputs do not enhance communication (1958). If indeed, humans have but one channel of communication to the brain, the addition of completely redundant text may not increase communication effectiveness.

Another possible explanation is that the interactivity of the computer-based program allowed students to pace their knowledge acquisition, independent of delivery mode. Several studies in the past that have demonstrated significant differences based on redundancy levels have been conducted on linear, program-paced materials (Baggett, 1984; Kozma, 1991). The presentations in the present study were entirely student paced--subjects could remain on each instructional screen as long as they desired. They could also replay the audio segments and back up to review previous instructional screens.

The content of the instructional program for this study was introductory material for the students. The material was, by design, low cognitive in nature. The majority of the questions on the pretest, posttest, and retention tests were recall or comprehension oriented. Very few questions required students to apply, analyze, or evaluate what they had learned. Consequently, the researchers raise the question about the relationship of content difficulty to media format. Future research may investigate whether or not more difficult content, designed at higher cognitive levels, would differentiate among students' achievement levels.

Results of this study also support prior research that demonstrated that subjects with higher verbal skills do not profit significantly from the addition of audio (Barton & Dwyer, 1987; Main & Griffiths, 1977). Perhaps students with lower verbal ability and lower reading scores would have demonstrated a significant difference with regard to redundancy levels.

Additional research questions related to student achievement might include: (a) Whether or not different content areas (such as grammar, literature, science, mathematics, or language) presented at various cognitive levels would produce different results; (b) The significance of implementing audio for poor readers or non-native English speakers; and (c) The relationship between students' preferred learning modality and the delivery mode.

Credits

This study was conducted through a Faculty Research and Creative Scholarship Award from the Division of Sponsored Research at the University of South Florida, Tampa, Florida.

References

Baggett, P. (1984). Role of temporal overlap of visual and auditory material in forming dual media associations. *Journal of Educational Psychology*. 76, 408-417.

Barron, A. E. (1992). *CD-ROM Tutorial*. [Computer Program] Tampa, FL: Florida Center for Instructional Technology

Barron, A. E. & Kysilka, M. L. (1992). The effectiveness of audio in computer-based education. Paper presented at the American Educational Research Association, San Francisco, CA, April 20 - 24.

Barton, E. A., & Dwyer, F. M. (1987). The effect of audio redundancy on students' ability to profit from printed-verbal/visualized instruction. *International Journal of Instructional Media*, 14, 93-98.

Broadbent, D. E. (1958). *Perception and communication*. New York: Pergamon Press.

Hartman, F. R. (1961). Single and multiple channel communications: A review of research and a proposed model. *AV Communication Review*, 9(6), 235-262.

Jasper, F. (1991). The relationship sound-image. *International Journal of Instructional Media*. 18(2), 161-174.

Kozma, R. B. (1991). Learning with Media. *Review of Educational Research*, 61(2), 179-211.

Main, R. E., & Griffiths, B. (1977). Evaluation of audio and pictorial instructional supplements. *Audio-Visual Communications Review*, 25, 167-179.

Muraida, D. J. & Spector, J. M. (1992). Toward effective use of speech in CBI. Paper presented at the Association of Computer-Based Instructional Systems. Norfolk, VA, November 8-11.

Nugent, G. G. (1982). Pictures, audio, and print: Symbolic representation and effect on learning. *Educational Communications and Technology Journal*, 30(3), 163-174.

Severin, W. (1967). Another look at cue summation. *A V Communication Review*, 15(3), 233-245.

Travers, R. M. W. (1964). The transmission of information to human receivers. *AV Communication Review*, 12(6), 373-385.

Simulation Training for Electronic Mail Systems

ANN E. BARRON & KAREN IVERS
Instructional Technology Program
University of South Florida, Tampa, FL 33620, USA

Introduction

The use of telecommunications is flourishing in education. Teachers have discovered that they can easily and inexpensively connect students with their peers in other parts of the country and the world. Many of the educators communicate through free-access networks, such as Bitnet or Internet. Other teachers choose to subscribe to one of the commercial educational networks, such as the AT&T Learning Network or GTE World Classroom. These commercial networks provide a more structured program than the free-access networks and focus on connecting students for interchanges that focus on issues of global interest (Barron & Orwig, 1993).

The benefits that telecommunications can offer to education are significant. For example, telecommunications is an excellent means to teach multicultural awareness. "Indeed, never before could teams of students, thousands of miles apart, engage in dialogue through which they jointly construct a model of their respective economics, cultural surrounding, or ecologies, and then collaboratively test its implications" (Salomon, 1991, p. 43).

Communications skills can also be enhanced through telecommunications. Students from different schools, nations, or countries can send their compositions for others to read, critique, and review. Several studies have reported a significant increase in the quality of students' writing with distant audiences (Cohen & Riel, 1989; Wright, 1991).

Telecommunications also inspires students and teachers and makes learning exciting and relevant. Studies have found it to be very motivating for students to correspond through telecommunications with experts who would be inaccessible through other means (Perry, 1984).

Telecommunications in Florida

In 1982, the Florida Legislature established the Florida Information Resource Network (FIRN) to enable administrators to transmit school records via telecommunications. FIRN is a wide-area telecommunications network that connects data centers and computer resources at universities, community colleges, and school districts throughout Florida (Schmid, 1990).

Through FIRN, public educators in Florida have access to a wide variety of computing resources and services. For example, teachers and students can retrieve information from remote databases at the state universities, including the online library catalogs and abstracts in the Educational Resources Information Center (ERIC). Teachers can also download a weekly lesson guide to NewsWeek magazine or image files of satellite weather. There is no charge for connect time to FIRN, and local dial-up and/or toll-free numbers are provided for modem access.

In 1991, the FIRN network was expanded to include a realtime electronic mail system, called FIRNMAIL. Through FIRNMAIL, educators can send and receive messages and participate in conferences of common interests. FIRNMAIL accounts are free to all public educators, and students can obtain access under the supervision of a teacher.

The Training Need

According to Paul McGinnis of the Florida Information Resource Network, over 6000 educators in Florida currently possess accounts for FIRNMAIL, yet less than 25% are active users. (An active user is defined as one who accesses the system at least twice a month.) One of the reasons for the minimal use is the lack of effective training methods. In the past, users generally learned how to navigate through the system in one of three ways: (1) By reading the manuals provided by FIRN, (2) By attending a training session with large-group instruction, or (3) By trial and error.

Although there are several technical training personnel located throughout the state to conduct training, it is difficult to include hands-on activities in the sessions. The training generally takes place in schools and very few buildings have more than one or two telephone lines for multiple access to FIRNMAIL. Most of the training in the past consisted of a demonstration of the FIRNMAIL system through a computer, modem, and large screen projection unit.

The need for a more effective vehicle to train educators to use electronic mail led to the development of a computer-based simulation. The simulation was designed to emulate the FIRNMAIL system without requiring telephone connections or user identification numbers. The program was also designed to enable educators to access the training on an individual basis at their own schools or homes.

Design and Development of the FIRNMAIL Simulation

The simulation was made to provide guided practice for the most common functions of FIRNMAIL. The topics were divided into four major categories, which became the simulation menu choices. These categories include: (1) Logging in and out of FIRNMAIL, (2) Creating and Sending Messages, (3) Reading and Deleting Messages, and (4) Indexing Messages. Although students are encouraged to access the menu selections in sequence, they can choose them in any order, repeat them if they wish, and exit at any time. Throughout the simulation, maximum learner control is provided with a pulldown menu and course map.

To enhance transfer, the simulation was produced with system screens captured directly from FIRNMAIL. The program also emulates the keystrokes of the system as closely as possible. Sound effects are included to simulate the dialing procedure, and visual effects provide realism for the screen displays.

Pilot Testing and Implementation

The simulation program was completed in January, 1993, and pilot tested at the University of South Florida. Initial reactions to the program were extremely positive. Comments include:

"It is very user friendly."
"Information on FIRN practical -- thanks."
"I like being able to read and learn at a slow pace."
"Great! Very informative and easy to understand."
"Walks you through the process."
"Fun and easy to use!"
"Very clear -- easy to use and understand."
"User friendly! -- even for elementary!"
"Directions very specific."
"It was very simple and understandable. This is very interesting for a novice such as I."
"Wonderful way to learn about the program."
"Very clear, very visual realistic."
"Easy to get around."

Although the program appeared to be well received, an empirical study was designed to test the effectiveness of the simulation against the alternate instructional strategies.

Review of the Literature

Research on simulations is controversial. Thomas and Hooper (1991) reviewed and categorized 29 simulation studies, most of which indicated no difference in knowledge gained when compared to other methods of instruction. They did find, however, that simulation groups presented a higher degree of transfer, citing Mayer (1981) that "the value of ... simulations appears to be greatest where the material to be learned is foreign to the learner and the goal of the instruction is transfer" (p. 500).

Other studies are contradictory. Kinzer, Sherwood, and Loofbourrow (1989) conducted a study where one group of fifth graders acquired knowledge about a food chain by reading an expository text, while another group of fifth graders used simulation software. The non-computer group outperformed the computer group on all measures. Woodward, Carnine, and Gersten (1988), however, reported significant differences of $p<.01$ on basic facts and concepts that were reinforced by a simulation treatment versus structured teaching alone. Much of the controversy and mixed results of simulation studies can be attributed to fundamental weaknesses in research design, the multiple definitions and subcategories of simulations, and the unknown quality of the simulations used (Thomas & Hooper, 1991; Woodward, Carnine, & Gersten; 1988).

Recommended considerations for the design, purpose, and quality of simulations include the level of fidelity, face validity, learner control, screen design, menu types, and other design characteristics (Alessi & Trollip, 1991; Aspillaga, 1991; Garhart, 1991; Milheim, 1991; Reigeluth & Schwartz, 1989; Ruben & Lederman, 1982; Schuerman & Peck, 1991). Expert analysis and pilot tests have demonstrated that the FIRNMAIL simulation has a high degree of fidelity, a design appropriate for the intended audience, a high degree of face validity, maximum learner control, comprehensible screen designs, accessible menus, and helpful feedback.

Investigation of Effectiveness

In order to test the effectiveness of the FIRNMAIL simulation, a study is being conducted at the University of South Florida, Tampa, Florida, during the Spring and Summer semesters of 1993. The population consists of students enrolled in Applications of Technology in Education, a two-semester hour class required for all undergraduate College of Education majors. The sample is being built on the basis of randomly selecting a minimum of 60 students.

Purposes of this study are to determine if the simulation strategy is an effective means for delivering instruction on an electronic mail system for teachers. In order to address this issue, answers to the following questions are being sought:

Achievement. Does the simulation program produce a statistically significant increase in knowledge-based achievement when compared to large-group instruction and text-based instruction?

Performance. Does the simulation program produce a statistically significant increase in skill-based performance when compared to large-group instruction and text-based instruction?

Time. Does the simulation program enable participants to perform common electronic mail functions in significantly less time than the large-group instruction and text-based instruction?

Perception. Is there a significant and/or practical difference in students' perceptions of FIRNMAIL with a simulation program when compared to large-group or text-based instruction?

Methodology

In order to provide three treatment groups necessary for this study, the subjects will receive instruction on the use of FIRNMAIL via three alternative strategies:

1. FIRNMAIL manual.
2. Instruction via large-group instruction with a "walk-through" of the FIRNMAIL system via computer projection.
3. Instruction via the FIRNMAIL simulation program.

Subjects will be randomly assigned to experimental groups--the three different instructional methods. The three groups will complete a pretest on two dependent variable--achievement scores and perception of FIRNMAIL. Each treatment group will then be provided with an identical verbal introduction and FIRNMAIL reference manual.

Following the administration of the independent variable--herein, one of the three instructional approaches--an achievement-based posttest will be administered to all treatment groups. Immediately following treatments, a post-experiment perception questionnaire will also be administered.

Following the posttests, all participants will be provided access to a computer and modem. They will also be given a temporary FIRNMAIL account and asked to complete a performance exercise which will include creating, sending, reading, and deleting messages in FIRNMAIL. Records will be kept to indicate the amount of time required to complete the performance activities and the accuracy of the performance.

The analyses will employ t-tests and one factor analyses of variance (ANOVA) with three treatment strategies (reference manual-only, instructor-led, and simulation). Dependent measures include pretest scores, posttest scores, time required for the performance test, and successful completion of the performance test. Significant mean differences will be analyzed via separate follow-up procedures utilizing the Least Significant Difference, when appropriate. In addition, descriptive findings and Chi-square analyses will be reported for the Student Perception Questionnaire. The level of significance, in all cases throughout the analyses, will be set at .05.

Summary

The State of Florida offers a free electronic mail system for all public educators. Although several training efforts are underway, it is difficult to meet the demand on a statewide basis. One problem associated with training has been the lack of instructional facilities with adequate telephone lines for online training.

One solution to this training need was the design and development of a computer-based simulation that could emulate the functionality of FIRNMAIL without requiring a telephone and modem. This simulation enables groups of users to be trained in a computer lab (without modems), and it also provides for individualized instruction at remote sites.

The purpose of the study is to compare the effect of a systematically designed simulation training program with the large-group demonstration. Results of this study will indicate if the simulation program has a significant effect on overall comprehension. In addition, the performance test will measure the degree of training transfer into the FIRNMAIL system.

If the results of the study indicate that the simulation program is an effective method for delivering FIRNMAIL training, similar programs may be produced for system training efforts. In addition, future endeavors will focus on statewide distribution of the simulation to provide access for all educators.

ERIC

References

Alessi, S. M., & Trollip, R. S. (1991). *Computer-Based Instruction.* Englewood Cliffs, NJ: Prentice Hall, Inc.

Aspillaga, M. (1991). Screen design: Location of information and its effect on learning. *Journal of Computer-Based Instruction*, 18(3), 89-92.

Barron, A. E., & Orwig, G. W. (1993). *New technologies for education: A beginner's guide.* Englewood, CO: Libraries Unlimited.

Cohen, M., & Riel, M. (1989). The effect of distant audiences on students' writing. *American Educational Research Journal*, 26(2), 143-59.

Garhart, C. (1991). Simulations: How real is real enough? *Journal of Interactive Instruction Development*, 4(2), 15-18.

Kinzer, C. K., Sherwood, R. D., & Loofbourrow, M. C. (1989). Simulation software vs. expository text: A comparison of retention across two instructional tools. *Reading Research and Instruction*, 28(2), 41-49.

Milheim, W. D., & Martin, B. L. (1991). Theoretical bases for the use of learner control: Three different perspectives. *Journal of Computer-Based Instruction*, 18(3), 99-105.

Perry, L. J. (1984, Summer). W.C.U. MicroNet: A state network for linking secondary science and math classrooms. *Journal of Computers in Mathematics and Science Teaching*, 16-18.

Reigeluth, C. M., & Schwartz, E. (1989). An instructional theory for the design for computer-based simulations. *Journal of Computer-Based Instruction*, 16(1), 1-10.

Ruben, B. D., & Lederman, L. C. (1982). Instructional simulation gaming: Validity, reliability, and utility. *Simulation and Games*, 13(2), 233-244.

Salomon, G. (1991). Learning: New conceptions, new opportunities. *Educational Technology*, 31(6), 41-44.

Schmid, B. (1990). FIRN: Florida's statewide educational network. *Florida Educational Computing Quarterly*, 2, 83-89.

Schuerman, R. L., & Peck, K. L. (1991). Pull-down menus, menu design, and usage patterns in computer-assisted instruction. *Journal of Computer-Based Instruction*, 18(3), 93-98.

Thomas, R., & Hooper, E. (1991). Simulations: An opportunity we are missing. *Journal of Research on Computing in Education*, 23(4), 497-513.

Woodward, J., Carnine, D., & Gersten, R. (1988). Teaching problem-solving through computer simulations. *American Educational Research Journal*, 25(1), 72-86.

Wright, W. (1991). International group work: Using a computer conference to invigorate the writing of your students. In W. Wresch (Ed.), *The English Classroom in the Computer Age* (pp. 100-103). Urbana, IL: National Council of Teachers of English.

ERIC

Learning Styles: Interactivity Levels and Path Control

ANTONIO R. BARTOLOME
Departamento de Didáctica y Organización Escolar
Universidad de Barcelona
Baldiri i Reixac, s/n, torre D-4°
08028 Barcelona, SPAIN

Some days ago I met a reputable, retired already, professor. He commented on computers:

- «Interactivity? Computers are not 'interactive', they can only to be 'reactive'. To be 'interactive' they must be 'intelligent'» -And it was implied that they were not.

In the afternoon, at a panel about Interactive Video Programs for Sciences Instruction at Higher Education, a german professor told me:

- «I have not seen yet one really interactive programme of Interactive Video».

At night I thought if my work was something that no longer holds. Are the current studies about Interactive Resources not more but a big fraud?

1 Does it exist the Interactive Video?

Actually, the key is in what is it meant by interactivity. Some definitions seem not to be very exigent: "The point is that interactive lessons require at least the appearance of two-way communication" (Jonassen, 1989).

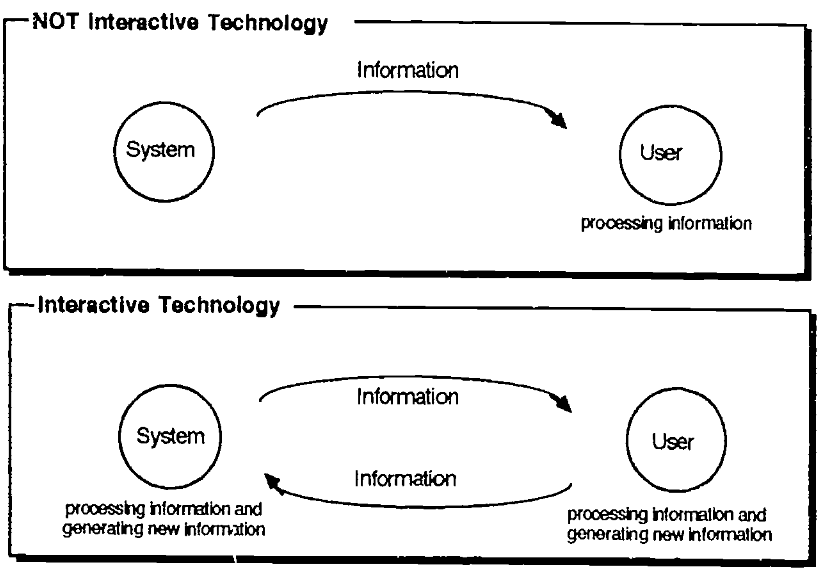
Other authors emphasize the active role by student: ". . . it changes the student from passive observer to active participant" (Anandam and Kelly, 1981). However, as Bosco marks (1989), they are really speaking of visible activity, because the student is always actively processing the information, even when he is listening to a lecturer or when he is reading a book.

As the old professor explained so well, the question about interactivity lies in the intelligence of computers. And for some ones, there is no doubt about this: "For the first time, a genuinely 'thinking' interacting technology, the computer, has become readily available to education" (Shavelson and Salomon, 1985). This kind of sentence started an interesting discussion with Kenneth Hawes: "Consider the following list of information technologies: the notebook and pen, the book, the library with its classification system and catalogues... Each of these is also an interactive technology" (Hawes, 1986). The answer of Shavelson and Salomon is short and clear: we need to enumerate important differences between the computer and other technologies and to examine the ways in which they are likely to affect the user (Shavelson and Salomon, 1986). And that is the biggest difference, found by Shavelson and Salomon: "Information technologies communicate with the user via one or more symbol systems"; that is, the ability of computer to represent the information in different ways, and to switch instantly between other representations.

And more: the control over the nature of the symbolic representation and the dynamic nature of some of these. But, *voilà*, seven years ago they were finding the strong link between "Interactivity" and "Multimedia".

As you can see, "Interactivity" can to be a word of uncertain meaning. And we need to offer our own definition. In this article, we mean by "Interactivity" in a human-machine communication process the fact that each end of the channel participates sending messages. They are received and interpreted by the other end, and this affects in some way in the next steps of the dialogue (Figure 1). The machine -the computer- can to run under sophisticated intelligent (?) systems or under extremely simple programs; that does not mind as the different cognitive level of subject.

Figure 1



The way as this communication is developed can be studied by means of qualitative methods, with categorized scales: lower levels should mean a smaller participation by, at least, one of the two end of the communication process. In the other side, higher levels should mean a very active participation of the two ends on channel in the dialogue.

2 Interactivity levels: the machine

Usually, four interactivity levels are considered (Hart, 1984; Priestman, 1984). These levels have been defined by the videodisc design and production group at the Nebraska University: they are a classification of videodisc players according to its interactivity degree. They are:

Level 0: Videodisc players with constant linear velocity

Level 1: Videodisc players with manual control

Level 2: Videodisc players with EPROM and microprocessor control.

Level 3: Videodisc players controlled from a computer

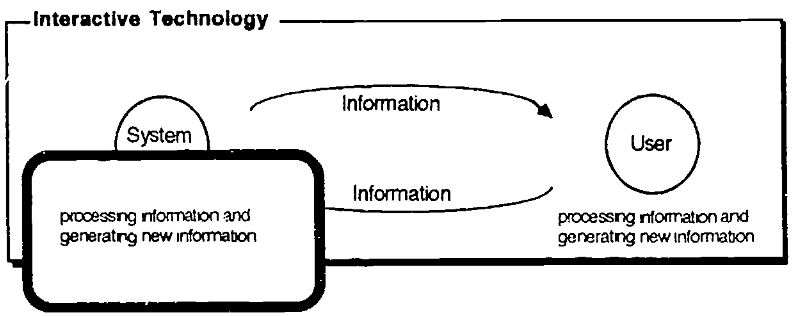
Some references to the fourth level design it as a "What more?", that, the other possibilities not included before.

These levels were defined according to the technical possibilities of the hardware, *not of the medium*. I think that this classification is not relevant from an educational point of view. The problem is a consequence of a bad interpretation of the Nebraska group work. They tried to prepare a classification of laserdisc players according to the interactivity features that they included. The classification was very easy for using and now it is frequently used as reference to interactivity levels. And this is not the same.

The Nebraska levels study the participation of the machine in this dialogue man-machine: the more features in the machine, the more interactive communication (figure 2). But it is immediate that the interactivity is also measured by the participation of the other end of the channel: the subject, the user.

We would like to remark an additional note: in fact, the actual interactive communication is not related with the hardware features but with the program design: I have used some courses based in very sophisticated machines but with a very little exploitation of the possibilities of the system.

Figure 2



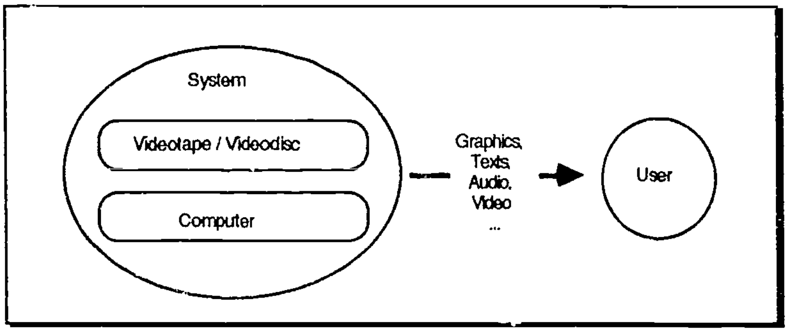
Nebraska levels are based in the possibilities of the system for doing this work.

3 Interactivity levels: the user

According to Salomon conceptions, we can begin assuming a system that supply audiovisual, iconic and verbal information (figure 3).

And we can assume that the System is able of locating almost instantly the information to present. The path control assignation, that is, who decides what information to show, it determines the interactivity level. Changing the point of view from the machine to the subject we accepting the previous Interactivity definition (Anandam and Kelly, 1981) and this Weller text (Weller, 1989) "...interactivity enables learners to adjust the instruction to conform to their needs and capabilities" (pg. 42).

Figure 3



From an evaluative research of educational software (CAI and Interactive Video programs), we prepared the next interactivity levels, also named "levels of system control by user". The next figure (Table 1) resumes these levels:

Table 1

Level 0 There is only one path, the same for every user. LINEAR	**Level 00** User can adopt a passive attitude **Level 01** User must to participate actively answering questions or participating in other activities
Level 1 There are several paths for different users. The paths have been built by the author of the programme. The user participates answering questions or doing activities; the programme selects the path according to the behaviour of user. BRANCHING	
Level 2 The user selects the information to receive. BRANCHING BY MENUS or DIRECT ACCESS TO DIFFERENT INFORMATION PACKAGES	**Level 21** User selects between options of program. **Level 22** User can to access to every information package of the programme, with its suggestions **Level 23** Direct access to every information.
Level 3 The user not only selects the information but also the code or symbols system.	

67

ERIC

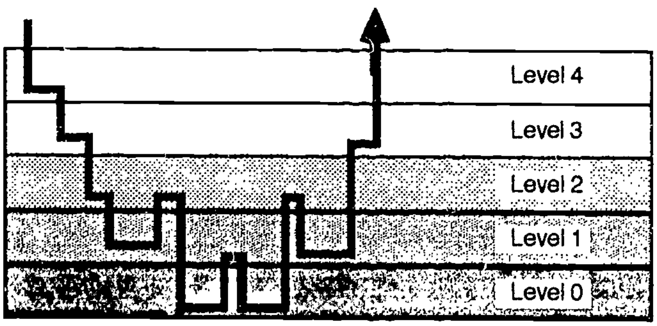
Level 4
The user selects also the source of information. He/she can access to external sources of information.

4 Which is the more adequate level?

This is not an easy question. Certainly, the theoretical conception of learning determines the answer. From a behaviorist perception, higher levels are a source of uncertain results during the learning process. By other way, from a cognitive perspective, higher levels offer more possibilities and more complex and beneficial learning contexts.

It seems that different levels are more adequate to different educational aims. It seems also that we can find a relation with the instructional level and the age. Other possibility is to consider that the user access to different levels during the programme (figure 4).

Figure 4



Some authors prefer without doubt a more active role for the student: the control of user over the learning strategy is the more efficient approach for the CAL designs (Hartley, 1981). And the Interactive Video programs must to supply the most amount of information because "the more information available, the more flexibility to join sequences" (Hosie, 1987, pg. 7). Nievergelt (1982) remarks that it must to avoid designs with a passive format: it has to be offered the maxim control of programme to students, as far as possible. The insistence on this aspect is explained because for some authors the limits of new media related with written texts is caused by the absence of opportunity for questioning him self, and this is related with the critics about students without control over the learning path. (Clark, 1984).

In this context, it was planned a research about interactivity levels at the University of Barcelona.

5 Interactivity Levels and Learning Styles: The research design

We wanted to study how students with different learning styles perceived interactive video programs, according to different interactive levels as they have been described here (see table 1).

Two versions of the same self-learning programme (Interactive Video) were distributed to students Educational Technology, at the University of Barcelona. Students used freely different Interactive Video systems, in Faculty rooms. Students did not know the existence of two versions of programme. Work was individual.

The two versions differed in how the subject accessed to videodisc and controlled the needed information: one version guided at every time the activity of user while the other version let him freedom to select the information.

Students used with the videodisc, a floppy disk. It contained the software of control and a specific software that registered activity process: number of sessions, when, used time for session and item, number of (and what items by session, which items he had visited... Also, the answers to questions and the answer to specific questions about his/her attitude.

Otherwise, attitude questionnaires were applied before and after the course. Students filled a Learning Styles Test, previously validated (Benedito, 1988).

Finally, we used some qualitative techniques for collecting information as interviews and group meetings.

We are not going to include here every conclusion from this research, but only someones that are referred to the exposed theoretical concepts. We do not include the statistics results or the data. You can ask this information to the author.

6. Conclusions

First, the version with a higher interactivity level was not the most accepted by students: they justified the results with the need of a sufficient initial knowledge of the contents for selecting the next information.

The freedom to select the path seemed to generate insecurity and the doubt, and this in students 23 years old.

Otherwise, no significant differences were detected in the way of working related with the different access to information. No differences about boring or tiredness, neither in time, number of sessions, error using the programme and other similar aspects.

Different learning styles did not significantly affect in the global perception of programme. We found some, significantly at 0.05 alfa level, correlations in partial aspects as analysis. However, and as we have said, the initial conception about interactivity was not confirmed.

A process analysis based in group interviews, gave these conclusions:

- Insecurity in front of the medium: for several this was the first time that they used a computer.
- Initial absence of knowledge of contents.
- High level of motivation for using the programme, independently of the interactivity design.
- Limited duration of work: an average of 4 hours, 20 minutes, distributed between 4 and 10 sessions.
- External problems for accessing to interactive systems.

Perhaps, the most relevant conclusion was a new model of interactivity that, at least for this programme, it seems very adequate. It is defined in this way:

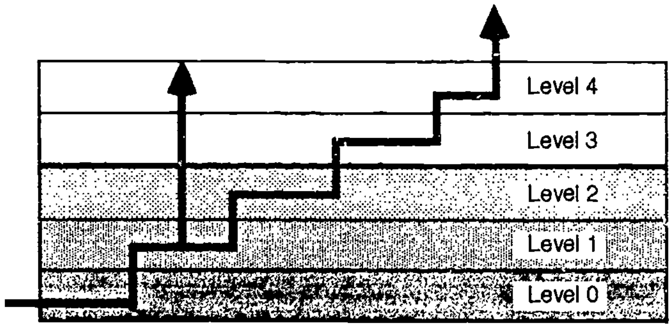
- To begin with low interactivity levels, so, the student need to attend only to the contend and the learning activities.

- To introduce gradually a higher capacity of control over the path, according to some limited options and on partially known contents.

- To offer, from the beginning, the option for a higher interactivity level for users that do not need or like a controlled way.

- To end with activities were the user has freely control over the programme (figure 5).

Figure 5



The problem of interactivity design is similar to the egg and the hen: it is not possible to take decisions if previously one does not know the domain of knowledge, but this is exactly the result of the program. So, in some way, it is possible to consider that this is the aim of an interactive self-learning programme: the user must be able of deciding how to continue his/her learning, or that he/she has reach enough maturity in the domain to work with a high level of Interactivity.

References

Acker, S.R. (1986). Redesigning the Human-Machine Interface for Computer-Mediated Visual Technologies. *Journal of Educational Technology Systems, 14*(1), 23-33.

Anandam and Kelly, (1981). *GEM. Guided Exposure to Microcomputers: An Interactive Video Program*. Miami: Miami-Dade Community College.

Asensi, M. (1987). Video-disco Interactivo - I. Cinevideo, (28), 40-44.

Benedito, V. et alt. (1988). *Innovación en el Aprendizaje Universitario*. Barcelona: PPU.

Bosco, J. (1989). Interactive video: Educational Tool or Toy?. In Bosco & oth.: *Interactive Video.* Englewood Cliffs: Educational Technology Publications Inc.

Clark, D.R. (1984). The Role of the Videodisc in Education and Training. *Media in Education and Development*, (Diciembre), 190-192.

Elliot, G. (1986). Interactive. The problem of design. Televisual, (Junio), 34.

Hart, A. (1984). Interactive Video. *Media in Education and Development*, (Diciembre), 207-208.

Hartley, J.R. (1981). Learner Initiatives in Computer-Assisted Learning. En U. Howe (Ed.), *Microcomputrs in Secondary Education*. London: Kegan Paul.

Hawes, K.S. (1986). Comment of "Information Technology: Tool and Teacher of the Mind". *Educational Researcher, 15*(2), 24.

Hosie, P. (1987). Adopting Interactive Videodisc Technology for Education. *Educational Technology*, *27*(7), 5-10.

Jonassen, D. (1989). Interactive Lesson Designs: A Taxonomy. In Bosco & oth.: *Interactive Video*. Englewood Cliffs: Educational Technology Publications Inc.

Nievergelt, J. (1982). The Computer-Driven Screen: An Emerging Mass Communications Two-Way Medium. *Educational Media International*, *1*, 7.

Nievergelt, J. (1980). A Pragmatic Introduction to Courseware Design. *Computer*, (Septiembre), 7-21.

Priestman, T. (1984). Interactive Video and Its Applications. *Media in Education and Development*, (Diciembre), 182-186.

Shavelson, R.J. y Salomon, G. (1985). Information Technology: Tool and Teacher of the Mind. *Educational Researcher*, *14*(5), 4.

Shavelson, R.J. y Salomon, G. (1986). A reply. *Educational Researcher*, *15*(2), 24-25.

Waltz, D. (1982). Artificial Intelligence. *Scientific American*, *247*(4), 118-133.

Weller, H. (1989). Interactivity in Microcomputer-Based Instruction: Its Essential Elements and How It Can Be Enhanced. In Bosco & oth.: *Interactive Video*. Englewood Cliffs: Educational Technology Publications Inc.

Withrow, F.B. (1985). Videodiscs: The Thinking Person's Audiovisual. *American Educator*, (Fall), 22-42.

This research has been done with financial support front the Minister of Education, CIDE, Spain.

ERIC

A multimedia editing environment promoting science learning in a unique setting–A case study

ROBERT J. BEICHNER
Department of Physics
North Carolina State University, Raleigh, NC 27695, USA

This case study examines how students used an extremely resource-rich environment to teach science to others. The purpose of the research was to observe how the students and their teachers approached their roles in this setting. A group of nine seventh- and eighth-graders attending a science magnet middle school were asked to develop an extensive series of information screens. These screens would be made available to the general public through the use of a touch screen kiosk. The school was located on the grounds of a large municipal zoo in the northeastern United States. The unique opportunities provided by the physical location of the school were supplemented by extensive technology facilities.

Background

In 1990 the school/zoo approached me for help in modernizing their text-based information screens describing current events and zoo exhibits. The computer coordinator suggested that this would be an excellent opportunity to exploit the unique setting by having school students create the screens for the zoo's kiosk. We began by creating a set of specifications for the customized editing software that would have to be created. We wanted the students to be able to easily manipulate multimedia objects–movies, sound recordings, scanned and electronically photographed images, colorful drawings, and text–so they could focus on what they were creating rather than becoming distracted by how they were creating it. A major consideration was that while the information screens had to be easy to create, they also had to be highly interesting and informative to zoo visitors. We were concerned at first that these might be mutually exclusive goals.

We also developed several questions about how access to this multimedia capability would promote science learning by students in this particular middle school. We wondered how the students and teachers would interact with the science content, the technology, each other, and zoo staff and visitors. We also wanted to examine how the students used their sources of information. Their requirements were quite unusual since they would be using multimedia to display what they had learned. On the other hand, the resources available to them were uncommonly rich.

Our first concern was with the ways the different people in the school/zoo setting related to each other and their surroundings. Studies of cooperative learning (cf. Johnson, et al., 1981) suggest that cognitive development is facilitated by peer interaction. The situations set up for students working on this project put a premium on the sharing of efforts and resources. Students almost always worked in groups of two or three when they were looking for information and creating multimedia screens. Vygotskian theory (Vygotsky, 1978; Wertsch, 1979) suggests that students' cognitive development can be enhanced by social interactions. People who are more competent assist learners by first helping or scaffolding their thinking and then fading assistance as the learner builds understanding and gains mastery of the material. The learner is an active participant in this process, which often takes place during problem-solving situations. The learner's understanding is mediated by his or her practical activity and existent knowledge. The data collected during this project describe situations where the computer coordinator, zoo staff, and even the students acted as "teachers." The observations indicate that it is also important to consider the visitors who used the kiosk. Although these people did not directly influence student

ERIC

cognition in the Vygotskian sense, they definitely seemed to have had an effect. Most teachers are familiar with the axiom that you learn more about a subject when you teach it than when you study it. This may be due to the review and organization of the content topics while preparing for presentation, a need to examine the material from a metacognitive viewpoint ("How is this best taught and understood?"), or simply seeing the material with more experienced eyes which are better able to see the connections between the content and other areas of study. Regardless of the mechanism, the students would be acting as teachers, accruing whatever educational benefits might arise from that enterprise.

The traditional understanding of constructivism has been an important tenet of educational psychology for some time. Even before Vygotsky's work, Piaget (1954) described his view of knowledge construction as a developmental process. This basic idea of learners constructing their own knowledge has recently been extended in a variety of ways. Scott, Cole, and Engel (1992) describe a cultural constructivist approach which assumes interactions between an active child and an equally active and usually more powerful adult. By itself, this is not much different from what we have already described. But their concept of cultural constructivism stresses that the interaction is mediated by cultural artifacts. In our case study the artifact includes a unique physical setting in conjunction with access to advanced educational technology.

Seymour Papert (1990) takes these ideas one step further and modifies the name of the theory itself to help explain what he means.

> We understand "constructionism" as including, but going beyond what Piaget would call "constructivism." The word with the *v* expresses the theory that knowledge is built by the learner, not supplied by the teacher. The word with the *n* expresses the further idea that this happens especially felicitously when the learner is engaged in the construction of something external or at least shareable...a sand castle, a machine, a computer program, a book. (pg. 3)

The key to what Papert is saying is the empowerment of students. They are given facilities to create something which has meaning to themselves and others. DeCorté (1991) noted that these "authentic activities" should be representative of future tasks and problems, provide many opportunities for social activities, and be rich in resources and learning materials. The project assignments met all these requirements. The students decided their work was similar to what television and magazine reporters do. To create these screens, resource materials had to be located and reviewed. Students had to select information from the extensive materials and facilities available to them, using their own criteria defining importance and interest. And they almost always worked in groups of two or three students, often debating the relative merits of different pieces of information. Awareness of the zoo visitors who would be viewing their screens seemed to be foremost in the minds of the student editors.

Description of the study

The setting

Approximately 65% of the 200+ students attending this urban science magnet middle school belong to recognized minority groups. Typical class sizes are 21 for seventh- and 29 for eighth-grade. The students, selected by lottery from a large pool of applicants, come from the local neighborhood or are bused in from the surrounding region. The school has won several awards for its programs and often acts as a model of how computer technology can be incorporated into various aspects of middle school education. In other words, the school appears to be a modern, highly desirable place for students to focus on science.

The self-named "Kid Krew" (from Kiosk Information Display Project) was a racially mixed group of nine seventh- and eighth-graders who each worked several hours per week on the K.I.D.

Project over the course of an entire school year, plus a few students who piloted the materials for the first year. Participating students were selected by the head computer coordinator on the basis of their reliability and availability. No special efforts were made to include or exclude students with academic abilities or deficiencies. Because of the technology-rich environment, all students had already attained a very high degree of computer literacy–probably reducing the Hawthorn effect commonly found in studies of instructional technology. There were two computer coordinators, three teachers, and half a dozen zoo and museum staff involved with the project at various times. A teacher training session held during the summer following pilot testing was attended by six teachers/support staff and was taught by both coordinators.

Resources

As noted earlier, state-of-the-art instructional technology was present throughout the school. Each classroom had at least one networked computer. There was also a lab of approximately 20 networked computers, CD-ROM drives, videodisc players, an electronic camera, sound digitizer, scanner, printers, a MIDI synthesizer, and a large collection of instructional and productivity software. Students were quite familiar with the operation of most of the technology. The editing software used by the students in the project was developed specifically for this research. Students also had access to a very wide assortment of commercial software located on file servers, CD-ROMs, and videodiscs, all present before the start of this project. Teachers were well trained in accessing these materials and used them often in their classes. Faculty members and the school administration often communicated by electronic mail. Students frequently visited the zoo exhibits and worked with the zoo staff. The school library included many materials dealing with animals including books, maps, and colorful photo-oriented magazines.

Software environment

The editing software is a hybrid of HyperCard and SuperCard. It also utilizes a commercial interface board supporting video display directly on the computer screen. By selecting items from the editor software's menus, students could work with an on-screen audio recorder (which looked like a cassette tape recorder), a video tool which functioned like their home VCRs, color painting and text tools, and a data linking tool for connecting pieces of information. Students normally used these tools to make touch-sensitive "hot spots" on the kiosk screen. By touching these areas, zoo visitors could see and hear animals, look for more information, or even print a handout sheet, complete with a map of the zoo and student-generated questions and comments about the animal on the screen.

The video tool (Figure 1) provides an example of how easy it was to create multimedia. Students used the VCR-like controls to operate the videodisc player until they saw the desired video segment appearing in the small "Video Screen" area. Clicking on the camera icon placed a "snapshot" from the videodisc onto the screen being created. Alternately, by use of the record button, a video sequence (with student-controlled audio) could be linked to a touch control automatically generated and added to the information screen. Later, when a visitor touched this spot, the movie would zoom out to fill the screen and play the video and audio as edited by the students. Similar tools allowed creation of strictly audio information, text, colorful drawings, and images captured from a scanner or electronic camera. Anything placed on the information screen could be moved, resized, and deleted through the use of a single set of keystrokes.

Methodology

In an effort to clarify my understanding of the situation, I utilized key informant interviews (in both formal and informal settings) captured on videotape and in field notes, along with videotaped observations of students, teachers, and support staff working together. The videotaped student editing sessions provided particular insight into the different types of student/teacher roles, the immediate and long-term goals of individuals and groups, and the participants' emotional responses to the situation. Situations where students interviewed zoo staff or talked to zoo visitors revealed the clearest view of what students saw as their role in the project. Detailed textual analyses of videotape transcripts made the research partly microethnographic, described by Levine (1990) as focusing on "interactional work that assembles systematic patterns of social behavior." The categories described below gradually emerged during this review of the data. The analysis began with the open coding methodology discussed by Strauss & Corbin (1990). Individual sentences were labeled as they were read from the transcript. The original labels were mostly descriptive in nature, including breakdowns as to whether a student was talking to another student or to a teacher, if students were referring to their audience, relaying a piece of information they had gathered from a resource, etc. These temporary labels were revised as more of the data was reviewed and a better impression of the central concepts emerged. For example, continued analysis led to a reclassifying of some of the teacher/student dialog into a new "teacher-directed action" category. This continuous category refinement eventually became what Strauss & Corbin call axial and selective coding. This can be described as making connections between categories and deciding on a "core" category which contains the overall pattern distilled from the data. The final results are discussed below.

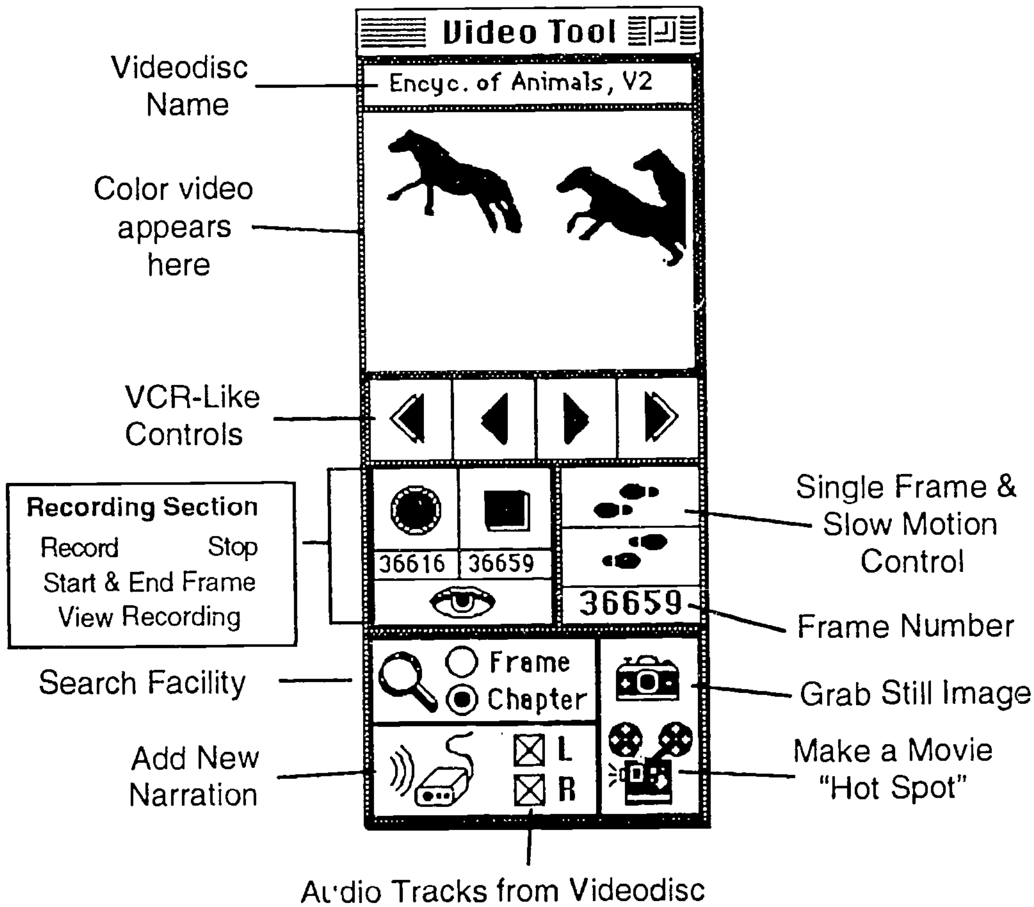


Figure 1. The Video Tool available in the multimedia editor.

Besides reviewing transcripts of student interactions, observations of teacher training sessions were utilized to provide clues as to what the teachers felt were the important educational aspects of the project. I also collected and studied sample screens, student handouts, and the reference materials used by the students. I addressed validity concerns by regularly comparing my observation notes of student activities with the computer coordinator's impressions. Since she worked with the students much more closely than I did, her comments were usually quite valuable. This also helped identify any observational biases my graduate assistant or I might have had.

Cognitive and Affective Impact

The final categories which evolved during the data analysis are detailed below. Writing for an audience emerged as the "core" category in the analysis of the videotape transcripts. Papert's (1990) extension of constructivism to constructionism seems particularly appropriate here. Student awareness of their audience and the importance of their task dominated the recorded data.

Writing for an audience

In this study we saw many instances of student concern for the people who would be viewing their multimedia information screens. This was perhaps most evident during the discussions students had while designing and creating new materials. "They [visitors] don't want to read a whole lot...just the main points."

This consideration for audience was also present for the non-textual pieces of information being assembled by the students. "You want a picture here. Just a little one. Just so they know what it is." Or similarly: "You need more close-up... . Is there a part where there's more of the animal? You can't tell what it is." Notice that one student is asking another student for an additional piece of information–not only showing their concern with the clarity of their presentation, but also indicating that the editing process was encouraging recall. The ability to edit also promoted reflection on the relative importance of content materials: "What should come on the screen first? Should we put our information first and then our pictures?"

Frequent debates over individual roles in the construction of new information screens indicated the level of student interest in what they were creating. Consider the following taken from field notes:

> The students begin planning several screens about the gemsbok. After some discussion, they decide to split their efforts, with one group at the computer editing station and the rest working with a videodisc player, looking for still images and video sequences. [At the computer, the following conversation was recorded:]
> Student 1: [while looking over a data sheet in the zoo keeper's manual] "I'll pick out the important stuff."
> Student 2: [sitting in front of the computer, actually entering some information] "No, don't put that!"
> Student 1: "Yes! It's important!"
> Student 2: "People don't want to know the [scientific] class!"
> Student 1: "Yes they do."
> Student 2: "Yeah! They'll go home and say 'Marge, did you know the class of the gemsbok is ...?"
> Laughing by both students. They quickly reached agreement that they would not use that particular piece of information. [About 10 minutes later as they were reviewing their nearly completed information screen:]
> Student 1: "Good, it goes along with what it said."

Student 2: "Anybody who's interested in the gemsbok..You got to think about the kind of person that would be..." [giggles]

So students were not only aware of their audience, but also of the importance of that awareness. They consciously made efforts to think about who would be viewing their creations.

Students also addressed the issue of navigation through their hypermedia database. This often led to revisions of their information screens or the links they made between screens. This can be seen in the following:

Student: "This whole kiosk thing...when it is done and down there...there is going to be so many different things that they are going to get lost in it."
Teacher: "That is the idea."
Student: "Because I mean, you get into one thing and there is no going back to the first thing...once you get off in a different direction ..."

This led students to make several modifications in the *existing* screens to make sure visitors could "back up" to earlier hypermedia nodes. Students created simple "Touch here to go back" hot spots. Once they had decided upon this course of action they included similar backlinks in all screens created from that point on.

Concern with accuracy of information

Students were as concerned with being accurate as they were with making the information interesting. At one point they were greatly distressed by the fact that an animal video did not exactly match what they recalled seeing in the zoo exhibit. In another session a student reading from a zoo keeper's reference book was heard saying "Let me tell you what to write so you don't mess up." This was actually received with no comment by the student at the computer keyboard. Evidently they had earlier reached some sort of agreement to work together in ways to minimize mistakes. Statements like this imply that placing correct information before the public was very important to the student editors.

Direction of students

A very definite reversal of student and adult roles was seen over the course of the study. During the first month or two, the computer coordinator and zoo staff often dictated both the content and layout of the information screens. When the students began using the technology, they were often told (probably in an effort by the coordinator and zoo staff to ensure accuracy and pleasing appearance) what information to include and how to incorporate it into an information screen. This was usually accompanied by instruction in how to carry out a particular task with the editing software. The students occasionally appeared to appreciate the help, but within a few weeks they demonstrated a strong desire to work on their own. Once they had mastered the editor, roles rapidly changed. Students not only picked out what information and layout designs they would use, they also began showing other students and even their teachers how to use the equipment and software. The computer coordinator mentioned that "Kids teaching other kids seems to work well." This growing confidence and expertise appears to be an outcome of the cooperative grouping established during the project. At first teachers directed the novice students. Then students of similar expertise worked together and added to their knowledge. Eventually these now-expert students worked in groups with students new to the project, "showing them the ropes."

It is important to realize that once the students became expert at using the technology, their attention focused on the science content they were trying to convey. The editing groups would have to review a sizable body of information before they could decide what was important enough to display. They would usually have to review several videodiscs, searching for appropriate

ERIC

images. Evidently they retained a great deal of this information. During their discussions students would often refer to a specific video image or piece of information they had seen that would be of interest to their audience.

Student 1: "Oooh! Look at this one!" (a frame from the videodisc)
Student 2: "That's enough! We already have three still frames to choose from!"
Student 1: "I found another good one!"

Duell (1986) noted that although identifying the main ideas about a topic and writing summaries of information are important study skills related to metacognition, these abilities are only gradually developed. The K.I.D. project tasks seemed to help students acquire these abilities fairly quickly. Students had few difficulties deciding on the most important concepts as they read materials and viewed video clips. We also noted transfer of some skills to other non-science classes. A teacher reported hearing two students talking about making a poster and what to include on it, saying that they "got the idea from the kiosk." While it is not clear if the students were discussing layout or content, it is obvious that they were applying what they had learned while working on the K.I.D. project to other settings.

Motivation

It quickly became obvious that both students and teachers were very enthusiastic with the project. The fact that the school was already inundated with technology implies that it may have been the project tasks that were so exciting. Teachers viewing the students using the system often exclaimed, "This is great!" Kid Krew students began skipping study halls and lunch periods in order to work on their screens. Often the computer coordinator would arrive in the morning to find students who had come in early and were waiting for her to open the door.

One of the few times that morale was low during the entire two-year K.I.D. project was when students encountered programming errors in the editing environment. When that happened, they quickly became highly frustrated. They also were annoyed when visitors looking at their screens did not understand how to use the system. This was especially evident when very young children placed their palms on the screen instead of using a fingertip to touch a hotspot. This caused the kiosk to behave in unpredictable ways and resulted in rather harsh operating instructions from the observing Kid Krew members.

Summary

The majority of student-to-student discussions dealt with creating information that would be meaningful and understandable to the general public. The editing process itself encouraged recall of content material, including the recognition and re-organization of the most important concepts in the content. Students saw the information they were collecting and presenting as having value and they constantly strove to create screens that were both accurate and interesting. They saw themselves as part programmers and part news reporters. As they became more familiar with the task, their work became much more self-directed. They began to teach others, including other students and the supervising adults, how to utilize the software. Students and teachers appeared to be highly motivated by the capabilities of the multimedia editing environment and by the job of creating information screens for the public.

Teachers in more typical schools, although probably not having access to as many technology-based materials, almost certainly have suitable holdings in their schools and community libraries. The point to understand is that it was not the unique setting that made the project successful. What seemed to matter is that the students saw that the work they were doing was important to somebody besides their teachers. It was worthwhile for them to learn new material and uncover additional resources. In the case of this project, those resources were extensive and decidedly "high tech," but I don't believe that was a necessary condition. The data demonstrate the importance of designing curricula that incorporate realistic, highly involving tasks. By establishing an environment where creative thinking about the content material is combined with real-world assignments, students will learn content, enjoy the learning process, and recognize that they have created something worthwhile–i.e. worth their time and effort.

References

DeCorté, E. (1991, April). *Research on learning and instruction: European contributions.* Paper given at a seminar for the Graduate School of Education, State University of New York at Buffalo, Buffalo, NY.

Duell, O. (1986). Metacognitive skills. In G. Phye & T. Andre (Ed.), *Cognitive Classroom Learning: Understanding, Thinking, and Problem Solving* (pp. 205-242). Orlando, FL: Academic Press.

Johnson, D. W., Maruyama, G., Johnson, R. T., Nelson, D., & Skon, L. (1981). Effects of cooperative, competitive, and individualistic goal structures on achievement: A meta-analysis. *Psychological Bulletin, 89*, 47-62.

Levine, H. (1990). Models of qualitative data use in the assessment of classroom-based microcomputer education programs. *Journal of Educational Computing Research, 6*, 461-477.

Papert, S. (1990). Introduction. In I. Harel (Ed.), *Constructionist Learning* (pp. 1-8). Cambridge, MA: MIT Media Laboratory.

Pea, R. (1985). Beyond amplification: Using the computer to reorganize mental functioning. *Educational Psychologist, 20*, 167-182.

Piaget, J. (1954). *The Construction of Reality in the Child.* New York: Basic Books.

Scott, T., Cole, M., & Engel, M. (1992). Computers in education: A cultural constructivist perspective. In G. Grant (Ed.), *Review of Research in Education* (pp. 191-251). Washington, DC: American Educational Research Association.

Strauss, A., & Corbin, J. (1990). *Basics of Qualitative Research: Grounded Theory Procedures and Techniques.* Newbury Park, CA: SAGE Publications.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes.* Cambridge, MA: Harvard University Press.

Wertsch, J. V. (1979). From social interaction to higher psychological processes: A clarification and application of Vygotsky's theory. *Human Development, 22*, 1-22.

This research was supported by grants from ComputerLand and Apple Computer, Inc.

Acknowledgments

This study would not have been possible without the cooperation of the zoo, teachers (especially Vicki Zimmer), administrators, and students. Special thanks go to graduate assistant Peggy Newport who made many visits to the school, fixed my programming errors in the editing software, and transcribed many, many hours of videotape.

Note: This article is based upon a more extensive paper already submitted for publication.

ERIC

Educational Hypermedia : from theory to practice...

BELTRAN Thierry
Laboratoire A.P.I., University Paul Sabatier / IUT A,
50 chemin des Maraîchers, 31 062 Toulouse, France.

This paper follows a set of publications concerning the study and the implementation of an educational hypermedia software, dedicated to Computer Aided Learning. The need of controlling, or at least supervising learner interaction in an Hyperdocument designed for teaching purposes has led us : • to define an architecture allowing the learner supervision and guidance (Beltran, 1991a & 91b), • to choose a model describing classical hypermedia systems so as to extend it in order to take into account the characteristics of the educational hyperdocuments defined (Beltran, 1992a & 1992c) and • to propose an object oriented approach for the designing and realization of educational hyperdocuments (Beltran, 1992b). This work is widely described in (Beltran, 1991c).

In this paper, we first present results pointed out concerning the theoretical definition of educational hyperdocuments and then we detail the adaptations and evolutions needed to put into practice the model we propose. These evolutions have been partly motivated by the realization of a first prototype: a hypermedia document to learn the three-dimensional structure of the human liver, from the study of the correlation of computed tomography (scan), sonography and gross anatomy. From the realization of this prototype, we try to broaden our analysis aiming at the definition and the specification of tools needed by the educational hyperdocument users. Then, we expect to draw 'guide-lines' and to establish conventions for designing these hyperdocuments.

For the clarity of the paper, we sum up in the following paragraph, the main results concerning the specification of the educational hypermedia software.

Educational Hypermedia Software : A survey of the theoretical aspects

We propose an architecture for an Hybrid system in which Hypermedia information can be combined with some directive 'sequences' in order to direct the learner in the information network. So, our aim is to design a system able to implement a wide range of educational applications (from the most directive one to the discovery environment) allowing, as far as possible, a 'hypermedia-oriented' interaction. With this purpose in mind, the first problem being studied is the supervision of the learner interaction.

The supervision of the learner interaction

Two systems compose the teaching / learning environment being studied : an hypermedia system dedicated to knowledge presentation and information seeking and, a control system made up of *Computer Aided Instruction modules* which ensure the pedagogical management at runtime and of *an interaction supervision module* (see figure 1). In (Beltran, 1991a) we have described the event-based communication mechanism between the presentation system and the control system. Some user-interface objects (the dialogue entities) have been designed to send events representing learner interaction. These events contain the results of the dialogue without considering the user interface details.

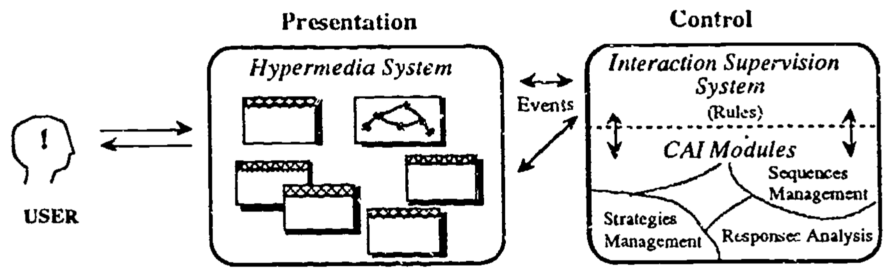


Figure 1. Educational hypermedia software general architecture

In our architecture, we have at the user interface level, Dialogue Entities which, by themselves, manage the interaction with the learner, and send events to the supervision module without ever activating a link directly. A dialogue Entity is made up of a set of *interactive elements* presented to the learner, plus a set of *virtual responses* available to the system (cf. figure 2). Each dialogue entity is created from a class in which the organization and the behavior of the interactive components are defined, i.e. the type of dialogue managed by the dialogue entity. So a dialogue entity manages the learner interaction according to its class (text scrolling, check box selection, object manipulation, selection of areas, text entry, ...) and sends events when needed. The response selection (by the system) depends both on the learner interaction and on the response tactic being used by the system. This implies the displaying of an area which contains comments (advice, explanations, ...) and links available to the learner.

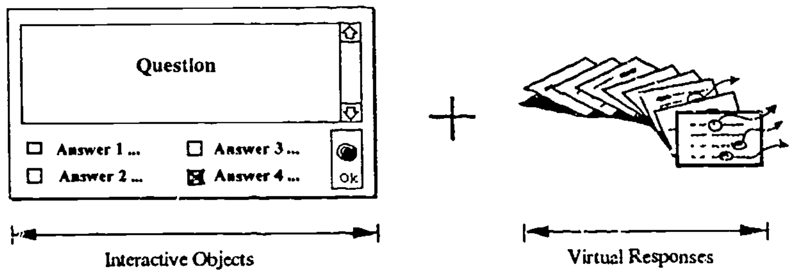


Figure 2. The structure of a dialogue entity

The following figure shows the object-oriented representation chosen to describe the dialogue entities (Beltran, 1992b). Only classes belonging to the lowest level of the hierarchy can produce instances ('multiple choice questions (mcq) with check boxes (cb) or scrolling list (l), able to accept from the learner a single answer (sa) or multiple answers (ma) : 'mcq_sa_cb, mcq_ma_cb, mcq_sa_l, mcq_ma_l', Interactive Areas, Indexes, ...); the other classes serve only to group and share common behaviors.

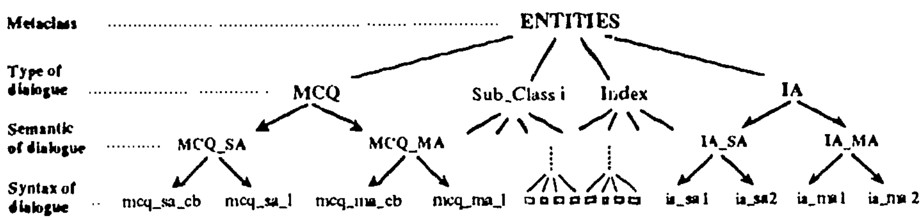
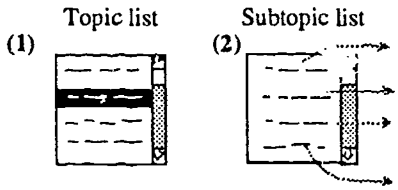


Figure 3. The dialogue entity class hierarchy

Example : the dialogue entity class 'Table of contents' with two selection levels, sub-class of the 'Index' class, implements a particular dialogue management (see figure 4). This class is used for making indexes, and particularly, for displaying and proposing a set of nodes and links matching a learner query.

The set of interactive objects is a scrolling list (1) in which each line represents a topic (a learner answer).



Each virtual response from the system, displayed after the selection of a topic, is a second scrolling list representing the second level of choice. This list (2) contains links to the selected sub-topic.

Figure 4. The 'Table of contents' dialogue entity

In fact, the event-based control system defines new types of links. These links are not 'frozen' in the hypermedia user interface and do not activate predefined nodes. They are represented by rules whose firing conditions are composed of events that may be received. We can distinguish between 'classical' links, unsupervised (i.e. learner-driven) and pedagogical links, which are under control of the supervision module. A classical link is used for free navigation through a limited domain (e.g. to explore information related to a specific topic). It can consist of several types: 'next', 'definition', 'example', 'detail', ... But, when the learner interaction has to be interpreted (e.g. to control the topical navigation, to analyze events sent by a dialogue entity), the author must define a pedagogical link. In (Beltran, 1991b), we have described several types of such links. Several rule-bases can be defined and each of them specifies a new behavior for the system. Thus, the definition of new types of links enables an author to determine when and how sequences must be learner-driven and system-controlled or system-driven and learner-controlled...

So far, we have described the architecture for the use of educational hyperdocuments. From this theoretical study we have pointed out the structures needed for the supervision of the interaction with the educational hypermedia software and we have defined a model extending the description of 'classical' hypermedia systems (Beltran, 1992c).

However, the implementation of such a system and the realization of a first application requires a more detailed analysis of certain aspects; for instance, the pedagogical functions of the system. With this purpose in mind, the following paragraph tackles the didactic role of the dialogue entities in regard of the different response strategies of the system and the means of structuring an educational hyperdocument according to pedagogical needs.

Educational Hypermedia system : from theory to practice...

Although the designing and realisation of hypermedia-based educational applications remains a subject of study, general steps seem to have emerged (Jonassen, 1990). For the development of our application, we have widely referred to (Safe, 1990a), (Safe, 1991a), (Safe, 1991b). But now, let us put into detail the context and the aim of this application.

Analyzing motivations and aims of the medical application

Although the vascular and segmental anatomy of the liver is well defined in the sonographic and computed tomographic literature, the three-dimensional relations of hepatic structures remain conceptually complex. This problem is interesting for us, because it concerns several actors which have different aims: • the learner in anatomy, which has only an approximate idea of the hepatic segments, • the teacher, which has to explain the complexity of this anatomy and • the radiologist, who needs a perfect knowledge of the hepatic anatomy to assist the surgeon in determining the feasibility of hepatic resection. Thus, the aim of this application is to help the user understand the liver anatomy, from pictures and schemas representing axial and sagital sections (gross anatomy), in correlation with radiologic imaging (MR imaging, Computed Tomography or scan, Sonography and Arteriography). The following scheme (anteroposterior view of liver) demonstrates the levels at which transaxial (a..d) and sagital (e..h) sections were collected, and the three intersegmental boundaries (planes A..C), defined by hepatic veins.

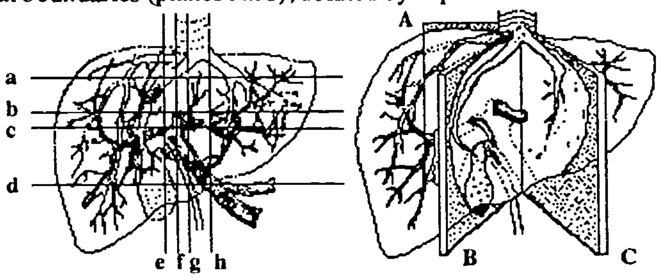


Figure 5. The different section planes

Thus, the hypermedia contains descriptive anatomical knowledge from which exercises are proposed. Moreover, the pedagogical functions of the system have to take into account learners of several levels (from novice to specialist) and several learning and teaching strategies. Emphasis is placed on the formation of a three-dimensional concept of hepatic segmental and vascular anatomy. The prototyping of this first application [1] allows us to acquire 'know-how' that can help us to specify general tools for the different users and 'guide-lines' for structuring educational hyperdocuments and later, specifying a designing method. Note: our aim is to study the pedagogical functionalism of the hypermedia system in order to put into practice our theoretical work; the mediatic aspects constitute for us a secondary problem, well-resolved in other works, e.g. (Höhne, 1992).

Dialogue entities and response strategies

When the interaction with the learner has induced the sending of an event, the supervision system receives the answer of the learner and has to return a response adapted to the strategy in use. B.Woolf (Woolf, 1992) defines four main response strategies in the context of the study of an Intelligent Tutoring System ('Brief', 'Verbose', 'Socratic' and 'Helpful'). Each of these strategies involves the activation of a set of particular **tutoring actions**, roughly similar to those we have defined to adapt the system responses when the learner interacts with a dialogue entity:

TUTORING ACTIONS
- *Concerning the question*
  - (1) Explain question
  - (2) Hints
  - (3) Reveal answer
- *Concerning the response*
  - (4) Explain answer
  - (5) Detail answer
  - (6) Offer advice

Figure 6. The tutoring actions

[1] In collaboration with the radiology department of the hospital of Rangueil in Toulouse.

In fact, the control system has to select one of the virtual responses that belong to the dialogue entity which has sent the event. This implies the displaying of a region corresponding to a tutoring action (explanation, advice, ...). This (multimedia) region can contain links proposed to the learner (for access to definitions, complementary paths or related topics).

In an educational hyperdocument, contrary to a 'classical' or 'intelligent' tutoring system, these functions can be activated as well as by the learner as by the system :

- When they are fixed during the designing stage, the different arrangements of tutoring actions allow the author to determine the general response tactic used by the system. Then, we find again the main strategies defined by B.Woolf for a tutoring system (informative, non-intrusive, directive, concise, coy, encouraging). In this case, the system will activate the tutoring actions corresponding to the current strategy as soon as a dialogue entity is presented to the learner. For example, in the case of the informative tactic, the system will systematically explain and detail its responses. In fact, the author determines the strategies to be used by means of a 'blackboard' tool.
- When knowledge concerning the didactic of the domain or the type of learner is insufficient in finding the best strategy, it is better to let the learner himself modify the behavior of the system, at runtime. Ttoring actions allow the learner to adapt the system responses according to his instant needs. Tacking this into account, we have defined two discourse control palettes, allowing the user to activate, when needed, a tutoring action applied to the dialogue entity currently in use (i.e., asking for the display of a specific virtual response).

By implementing such a system, we have provided the author with a new flexible tool in which the teaching - learning processes depend on the degree of control, in ways that we are only beginning to discover: today, there is neither designing method, nor rules providing a framework for the author during the realization of an educational hypedocument. In the same way, the choice of logical organization of the document remains related to the author's intuition.

The logical parting of the educational Hyperdocument

In the case of the application being developed, the users have much knowledge concerning this domain. This has led us to privilege a structure allowing quite an important degree of freedom. In this purpose, we have designed three main modules (Theoretical knowledge, Examples, Practice) which may be consulted on request.

a) Course, Examples, Practice :

Course : contains (multimedia) information related to the main topics. Several complementary paths are associated to each topic. The pedagogical links are important because they dynamically adapt the complementary paths according to the characteristics of the session consultation. In the case of the on-going application, this partition contains knowledge (grouped by topics): position in regard to the other organs, general surface views, 'see-in' diagrams, ligamentar, vascular and segmental descriptions, graphical animations, terminology, 'french-english' equivalence (text and sound), abbreviation list, ... This partition is divided into two interrelated modules. The first one corresponds to a topical navigation (undetailed) and the second one contains sets of details that can be easily accessed from the first one. Each topic begins with a presentation and ends with a summary. **Examples** : This partition contains concrete or simplified cases related to the topics of the course, and 'classical' links to series of examples (previous, next, ...). The pedagogical links enable the system to build a learning path through several examples, from a specific topic in the course. **Practice** : Most of dialogue entities of the educational hyperdocument are stored in this partition. Multimedia presentations are provided for the learner, and problems managed by the dialogue entities are proposed to him. The virtual responses from the system are hidden and activated according to the learner interaction and the system strategy, the on-going application, this partition contains problems of several types: ligamentar, vascular and segmental recognition exercises, terminology exercises, spatial recognition exercises, manipulation exercises (rotations, ...). Instances in both example and practice displays are divergent, range in difficulty, are presented in an easy-to-difficult sequence, and include a variety of representation forms.

The need to let free the learner in document structure has led us to provide increased support for searching and browsing through an educational hyperdocument.

b) Data access : browsing and searching facilities

The '**learning browser**' enables the learner to activate the different nodes related to a topic. This tool, adapted to the logical structure of the document (course, examples, exercises), only activates the different windows of the three partitions. For example, the learner can search for the information he needs, and skip to an example display before resolving an exercise.

The '**topic browser**' allows • the navigation within a partition, • the displaying of indexes and • the access to the query tool. Indexes and maps allow direct access to needed information and selection of paths. However, the author can limit the number of links proposed by the index, specifying the topics that can be consulted in a particular context. In this purpose, the author has to use a dialogue entity of the 'Table of contents' class (see figure 3).

The query mechanism, presented in (Beltran, 1992c), is based on the formal representation discussed in (Halasz, 1990), (Safe, 1990b). We have presented a prototype (using Oracle and HyperCard) allowing the user to express graphically SQL queries (see 'Query Builder'). The result of a query (list of nodes) is stored, by the 'Index Builder' in a dialogue entity of the 'Index' class.

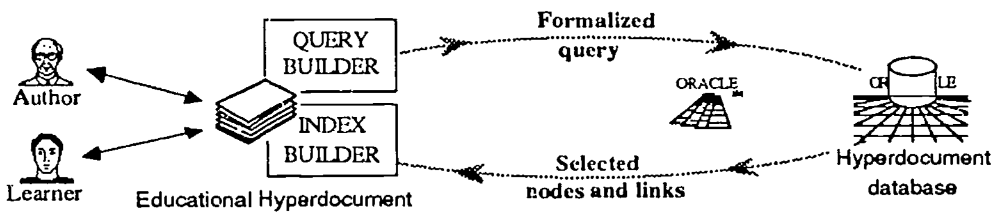


Figure 7. The interactive indexing facility

So, an index can be activated as well as directly as in response to a query. It can be activated for two main reasons : • for the consultation of related topics (by means of a restricted 'Table of contents' dialogue entity), • for a topical navigation (in this case, a full index is activated and the rule-base used by the supervision system is changed). Several works have pointed out how important retrieval facilities are for hypertext beside navigation (Cousins, 1989), (Hofmann, 1992). In the case of an educational hyperdocument, it is advisable to moderate this assertion according to the degree of freedom wanted by the author and the competence degree of the learner : data access by browsing (exploring) is prefered by a novice learner whereas data access by searching is more convenient for experienced learners. In this case, query facilities become necessary, especially to find good starting points for a more detailed investigation, i.e. interactive browsing. Thus, the author must have the possibility to forbid the learner to use the browsers during certain sequences. In this way, the freedom degree of the learner is diminished (the browser palettes are hidden). For example, it is possible to impose an essentially descriptive path (restricted to the 'Course' and 'Details' partitions) or a purely practical one (restricted to the 'Exercises' et 'Practice' partitions).

The on-going application, developed using HyperCard (on Macintosh computer), allows us to refine the theoretical model from information gained through testing from a particular case (particular didactic functions of the dialogue entities and particular structuring of the educational hyperdocument). Analysing the prerequisites and features of the application being developed, and considering the practical choices that have been made, we hope to gather information and to acquire a more general 'know-how' applicable to other realizations. In the following paragraph, we analyse the first experimental results.

Educational Hypermedia system : some practical results

(Safe, 1991a) describes, in fourteen steps, the development of a hypermedia database, from the designing stage to the CD-ROM production stage. This technical classification, taking into account the needed tools for each different medium, is necessary but not sufficient: a detailed analysis, from the pedagogical point of view, of the first steps of the designing stage is lacking. When designing an educational hyperdocument, three essential problems need to be solved : • what degree of freedom allowed to the learner in the system, • what are the teaching strategies to be chosen, • what is the structure or organization suitable for the document ? The three problems are related, yet they seem also dependant on indications of the context such as the pedagogical aims, user profile and the domain concerned... So, it is difficult to evaluate the impact of changing those parameters as we cannot generalize from only one experience. Although the structure we have chosen may be adapted to other domains (since three generic levels are taken into account: theoretical level, descriptive level and experimental level), some problems shall need a radically different approach. For example, teaching the reasoning processes (in mathematics, medicine, management, ...) forcibly lead to a different document organization and to the implementation of new didactic functions.

Nevertheless, the analysis of our approach can help us to master the parameters which have an influence on the three previous points: we have presented different means allowing an author to modify the supervision level in an educational hyperdocument. First, as early as the designing stage, the pedagogical links allow the definition of a wide range of supervised paths. The dialogue entities allow the management of a more or less complex interaction with the learner, while observing a fixed pedagogical strategy. Finally, even when the hyperdocument has been realized, the author has the possibility to act on the freedom degree of the learner, allowing him the use of the control tools presented in chapter 3 (the document structure browser, the index access tool, the tool controlling the tutoring actions).

To summarize, the author 'weaves' an information network, more or less directive, more or less supervised, and then, decides to allow the learner to be more or less autonomous.

Critics and Prospects

The current realization allowed us to implement a first experimental hypermedia-oriented educational software, based on our previous theoretical studies. In this way, we have tested the generic tools presented in (Beltran, 1992c) (dialogue entity editors, pedagogical links editors, ...) and developed new tools, although they have been specifically designed for a particular class of applications (learning browser, control palettes). From this first practical experience, we expect to make the theoretical model evolve, and we hope to draw a framework for the designing of educational hyperdocuments...

Concerning the query mechanism, based on the formalisation of educational hyperdocuments, several interesting issues can be discussed : • For the author: help services for the designing and realization stages can be defined. For example, it can be helpful to know the number of dialogue entities that belong to a predefined path or that contain a specific word, ... • For the learner: such a

ERIC

system can provide powerful tools for information seeking. • For the system: this approach constitutes a new step towards the cooperation between Intelligent Tutoring Systems (ITS) and Hypertext systems. The formalization of educational hyperdocuments provides a basis from which an ITS can build learning paths according to its models.

In conclusion, despite our limited experience, we have proved that our work concerning the definition of educational hypermedia software, is flexible enough to implement educational environments of several freedom levels. The tools described provide a test-bed for exploring new issues and testing new teaching and learning functionalism. But, obviously more experience is needed before drawing a designing method adapted to the educational hyperdocuments. However, this first stage was necessary, at least in order to confront the theoretical model with the practical reality...

References

Beltran T. (1991a). Une architecture pour le guidage de l'apprenant dans un système Hypermédia éducatif, *Proc. of "Hypermédias et Apprentissages'*, Chatenay-Malabry, 24-25 Sept., pp. 95-110.

Beltran T. (1991b). Hypermedia and Learning: an architecture for hybrid systems, *Computer Aided Learning and Instruction in Science and Engineering*, Lausanne, 9-11 Sept., pp. 167-176.

Beltran T. (1991c). Etude d'une interface Multimode pour la production d'Hypermédias éducatifs par l'Atelier de Génie Didacticiel Intégré, *Doctorat de l'Université Toulouse III.*

Beltran T. (1992a). A formal representation for Educational Hypermedia software, *Proceedings of the 9th Int. Conf. 'Technologie and Education' : ICTE'92*, Paris, 16-20 March, pp. 1241-1245.

Beltran T., Péninou A. (1992b). An Object Oriented Approach to produce Hypermedia Educational Software, *Proceedings of the 4th Int. Conference 'Computers and Learning' : ICCAL'92*, Wolfville, Nova Scotia, Canada, 17-20 June 92, pp.111-123.

Beltran T. (1992c). Hypertext, DataBases and Computer Aided Instruction : where is the match ?, *3rd Int. Conf. ' 'Databases and Expert Systems Applications'*, Valence, 2-4 Sept., pp. 215-220.

Cousins S.B. (1989). Information retrieval from hypertext : update on the dynamic medical handbook project, *'Hypertext'89' Conference*, Pitsburgh, 1989, pp. 199-212.

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

Hofmann M., Schmezko S. (1992). Graphical Structure-Oriented Search in a Hypertext System, *3rd Int. Conf. 'Databases and Expert Systems Applications'*, Valence, 2-4 Sept., pp. 179-184.

Höhne Dr. (1992). Three-dimensional medical atlas, *1st European Conf. and Trade Show on Educational Technology, (LearnTec'92)*, Karlsruhe, Germany, 10-12 Nov. 1992.

Jonassen D.H., Mandl H. (1990). Designing Hypermedia for Learning, Computer and Systems Sciences, *vol. 67, NATO ASI series, Proceedings of the NATO Advanced Research Workshop on Designing Hypertext/Hypermedia for Learning.*

Safe (1991a). HYP/19 Final report on authoring hypermedia material, *DELTA project D1014: Standard Authoring Facility Environment (SAFE).*

Safe (1990b). HYP/20 Final report on hypermedia databases, *DELTA project D1014: Standard Authoring Facility Environment (SAFE)*, 20 Nov. 1990.

Safe (1991a). HYP/21 Final report on user interfaces for Hypermedia, *DELTA project D1014: Standard Authoring Facility Environment (SAFE)*, 20 March 1991.

Safe (1991b). Final report on the SAFE project, *DELTA project D1014*, 15 June 1991.

Woolf B. (1992). Building Knowledge-Based Tutors, *4th Int. Conference 'Computers and Learning' : ICCAL'92*, Wolfville, Nova Scotia, Canada, 17-20 June 1992, pp. 46-60.

Sound in the Multimedia Interface

MEERA M. BLATTNER
Department of Applied Science
University of California, Davis, and
Lawrence Livermore National Laboratory
Livermore, CA 94551, USA

"Everything we do is music."
John Cage

Multimedia can have a significant impact on education. We must create appealing interfaces that draw students into their work, challenge them, and provide them with a full range of intellectual experiences. A learning environment that is richer and more enjoyable than those we now have available can be created by the incorporation of many types of sounds into the computer-user interface. New research directions in audio are emerging driven by the availability of software tools for sound and the excellent sound capabilities in modern workstations. In the material below, we show some new ways that audio can be used to create interesting sensory environments stimulate and engage the minds of students.

The focus of this article is on recent developments in audio, however, the motivation for the use of sound is to provide a richer learning experience. To accomplish this we must tie it into the emotional and intellectual effects of sound on the user. This article begins with a description of the flow state, that state of mind in which we work deeply involved with what we are doing, and propose some techniques for achieving the flow state through our use of the user interface.

We can hardly imagine a multimedia interface that does not make use of sound. Surprisingly, sound is a medium that has been slow to understand and develop in computer interfaces. Early systems had blips and bleeps to call the user's attention to the screen, usually because of an error or the need for immediate action. The standard current uses of sound in multimedia interfaces incorporate recorded speech or add composed music to scenes. In the future, we can expect to see sound used in entirely new ways. In the section on audio and mental engagement there is an overview of the different types of sound we can use in our interfaces. Nonspeech audio is of particular interest for two reasons: 1) it is easy to use with our workstations and home computers, and 2) it is an interesting new way to use sound and communicate information. Auditory messages called *earcons* were designed to provide a more structured approach to the use of sound than blips and bleeps. Earcons are constructed from short sequences of sounds (motives) that can be compounded in three basic ways to create larger more complex units. They can be used on any computer that has audio output. Earcons have been tested and found to be effective in the interface. A section of this article is on the construction of earcons followed by the concluding remarks.

The Flow State

The *seductive interface* is a concept introduced by Timothy Skelly (1991), a designer of video games. To be seductive, an interface ideally engages both the mind and the emotions of the user. Seductive interfaces temporarily disengage us mentally from the "real world" by presenting the mind with alternate, engaging activity. Seductive interfaces involve drama and suspense as well. A truly seductive interface is self-teaching and highly conducive to learning because intellectual challenge must take place if the user's interest is to be maintained. The ideal self-teaching interface provides a set goals, challenges, and feedback for learning critical skills.

We often associate engagement and drama with leisure activities, such as TV and movies. Research using the techniques of the *experience sampling method* (ESM) has brought to light some interesting results. Surprisingly, leisure and mental engagement in many ways are opposite experiences. Leisure is not as uniformly enjoyable as is generally assumed. The most positive experiences in people's lives seem to come more frequently from work than from leisure settings. People do need to recuperate from the intensity of work in low-intensity, free-time activities. This seems to be culturally dictated, because in some societies after a full day's work leisure time may be spent playing musical instruments, carving, weaving, etc. The overall quality of life could be improved if people were aware of the negative feelings they had when their free time does not meet the conditions of flow. Secondly, if they realized the positive

ERIC

feelings they had while they are working, they could disregard the cultural mandate against liking work. (Csikszentmihalyi & LeFevre, 1989)

The flow state, the episodes when life is heightened and one one is deeply involved and mental energy is highly focused, is in many ways the opposite of the viewing experience. In flow experiences, people report very high concentration but ease of concentration--they feel active, strong, and in control. Concentration is so focused during flow activities that people typically report a diminished awareness of their surroundings and they lose track of time. The state is also me likely to occur when there is rapid, positive feedback about how one is doing in an activity.

Flow tends to occur when persons are confronted, or choose to confront themselves, with demanding opportunities for action (or challenges), which they feel capable of matching with their skills. High challenges and low skills typically result in anxiety, whereas high skills and low challenges cause boredom--and often anxiety, as well, when boredom is great enough. Flow activities involve a near equal pairing of high challenges and high skills, but the higher the skills and challenges, the more likely it is that flow will occur. Flow requires a clear cut goal and a specific task. Another important aspect of flow is that there is feedback. Passive experiences, such as watching TV, can cause guilt and discontent. With TV there is generally a pairing of low challenges with very limited use of skills. Thus people usually do not concentrate very much and they feel relaxed and passive. Viewing can be very involving and associated with high affect, but only three percent of the television viewing experiences are of this type. (Kubey and Csikszentmihalyi, 1990)

ESM research with adolescents has shown that video game play is quite activating and involves much higher reports of affect than does TV viewing (Kubey & Csikszentmihalyi, 1990). Video play is highly challenging, requires skill, and offers rapid feedback and thus possesses all of the key structural elements necessary to experiencing flow. Many video games are also programmed to increase in difficulty as the player's skill increases. To provide seductive interfaces we need to apply some of the techniques used by the highly successful entertainment industry for drawing us to the computer, and then devise clever ways to keep the student challenged and engaged.

Audio and Mental Engagement

To provide an interface that maximizes the learning experience for the student, it must:

- *Entice:* draw to the interface by curiosity and appeal,
- *Engage:* engage with challenging tasks and feedback, and
- *Enrich:* provide a rich sensory environment.

Audio is probably the interface designer's most useful, and at the same time, most under-used tool (Skelly, 1991). Audio is one of the most powerful methods of engaging the mind and providing information.

We need only to look at education in many places around the world to see how music is used to create flow experiences (Blattner & Greenberg, 1992). In many cultures, music is integrated with drama, dance, poetry, and art. Students learn through participating in these events, that is chanting, singing and dancing. Through these activities the cultural history is passed on with stories, ethics, history, and other cultural products of value. Western music is only thought of in aesthetic terms, but even in western music advertising and pop music are used to convey information. Music is used in ritual in almost every culture (Nettl, 1983). Religious music is frequently used to create a state of "other awareness," catharsis, and sometimes, frenzied and ecstatic states (Herendon and McLeod, 1982). It is thought that this is partly because of its hypnotic effect.

A range of agitated feelings can be created through pronounced and insistent rhythms (Rosenfeld, 1985). In many cultures, people use extended sessions of singing, chanting, dancing and drumming to induce altered states of consciousness. Scientists have proposed that certain types of drumming may produce powerful effects by driving the brain's electrical rhythms. This is similar to "photic driving" in which rhythmically flashing strobe lights can impose their rhythms on the brain. Music therapy is routinely used in care centers (Gilman & Paperte, 1952). Studies show that shoppers in stores with piped in music bought more items when slow music was played rather than music with a fast beat (Rosenfeld, 1985).

In the paragraphs below, we will look at some of the types of audio that can be used in the computer interface. With each of these, we will examine how using that type of music may be used to entice, engage, and enrich the quality of educational software.

89

Music: Music engages listeners with its rhythms, melodies, and interesting timbres. Care must be taken when music is used in programs that are used frequently, because music can be annoying if the same piece is heard repetitively (Blattner, Sumikawa and Greenberg, 1989). A technique often used in commercial programming is to use snippets of music to spark the viewer's interest and tie component parts together. Music specific to particular cultures is used in the study of history, geography, and anthropology.

Intricate rhythms and harmonies are designed to hold us spellbound. Music serves as a kind of cohesive, filling in empty spaces in the action or dialogue. Film music builds a sense of continuity, uniting the visual parts, which is important because theatrical film is like a jigsaw puzzle. The color and tone of music can give a picture more richness and vitality and pinpoint emotions and actions. Thomas (1973) says that music comes to bear in helping to realize the meaning of the film, in stimulating and guiding the emotional response to the visuals. A scene placed in a geographical context may be enhanced by local music. It is the ability of film to influence an audience subconsciously that makes it truly valuable to the cinema. Music can complete the total picture and produce a kind of dramatic truth, which the visual element is not always capable of doing.

Speech: What can be more enticing than a computer that speaks to you, even calling you by name? For the most part, however, speech is used for the communication of detailed and specific information. It is through speech (rather than through other sounds) that we communicate precise and abstract ideas. Speech will be used as input as well as output in the computer interface and speech recognition interfaces will be marketed for general use in the near future. Recent advances in speech recognition systems have made it possible to use a natural speech style and allow casual users to work easily with a speech system (Rudnicky, 1992). In spite of this, very little is known about building successful speech interfaces for two dimensional displays let alone three dimensional interfaces.

Virtual classrooms with students in different locations sharing their experiences will be a reality soon. Telepresence and teleconferencing networks will transmit images and sound from remote sites. Children in a French lesson in Chicago will converse with their counterparts in Paris. The applications of sound localization by NASA has shown the effectiveness of separating voices in space to improve their clarity (Wenzel, 1992). Scientists at Nipon Telegraph and Telephone (Miyoshi & Koizumi, 1992) have been studying the three-dimensional acoustic properties of teleconferencing systems to filter out extraneous sounds, enhance desirable sounds, and provide wide listening areas. Networks will allow collaborative engineering with the use of hand-gesturing, voice-assisted manipulation, grasping and manipulating objects (Weimer & Ganapathy, 1992).

Real-world sounds: These are the natural sounds of the world around us, such as leaves rustling or birds singing, or man-made sounds such as machine noises or even a band playing in the background. What about sound in our everyday life? Anyone who is interested in the sounds of everyday life should read the work of R. Murray Schafer (1977). Schafer describes "soundscapes," historical reconstructions of the sound that surrounds people in various environments. Examples are street criers, automobiles, the crackling of candles, church bells, etc. How better to entice students to study their history lesson that day?

An example of how real-world sound is used for feed-back is seen in a golf game by Incredible Technologies, the image resolution was too low to provide a visual cue when a ball was hit into the branches of a tree, instead the sound of leaves rustling and a bird squawk illustrated the event (Skelly, 1991).

Sampled sounds are recordings of real-world sounds, while synthesized sounds are made on the computer. Gaver (1993) lists the difficulties of using sampled sound for reconstructing real-world sounds: i) it is difficult to capture actual events because sounds are colored by the technology used to record them, ii) shaping recorded sounds relevant for auditory icons is difficult because available software is designed for music, iii) real-time modification is limited, and iv) the amount of memory required is often prohibitive. Nevertheless, there are many advances being made in synthesis algorithms in the fields of signal processing and audio engineering. We can expect to see much better tools for sound synthesis of real-world sounds in the near future.

Three dimensional sound: Three dimensional virtual sound is not really a type of sound, but rather the simulation of location and direction of virtual sound sources. We include this as a separate category because of the newness of the subject and the importance of this technology to the future of user interfaces. Wenzel (1992) separates sound sources into the three spatial dimensions by modeling the pinnea of the outer ear. A real-time digital signal processor, a Convolvotron, was designed by Scott Foster of provide filters that simulate the outer ear model. Instead of filters, spatial effects such as blurring, thickening, peaking, distancing, muffling, thinning, distortion and self-animation are created by Ludwig, Pincever and Cohen (1990) to give the illusion of three dimensions.

The immediate use of three-dimensional sound is in virtual reality, however, the appeal of sound surrounding the user is so great that we can expect to see this new technology permeate educational and recreational software. Music is enormously richer in three-dimensions, conversations are clearer when separated in space, and the sense of immersion in an artificial world is greater when sound surrounds the listener.

Auditory displays: Auditory displays are largely concerned with the interpretation of data into sound, such as the association of tones with charts, graphs, algorithms or sound in scientific visualization. These auditory display techniques were used to enable the listener to picture in his or her mind real-world objects or data. the intention of the first scientists that used this new technique was to provide another sensory dimension to the analysis of data.

Some of the first examples of work in this area are from the early 1980s. Bly (1982) interpreted points in higher dimensions to sonic equivalents, that is, sonic attributes were interpreted as dimensions. Mansur, Blattner and Joy (1984) translated points on an x-y graph into sonic equivalents with pitch as the x-axis and time on the y-axis (a nonlinear correction factor was used). Morrison and Lunney (1991) presented analytical chemistry data represented by sound to visually impaired students; Yeung (1980) also used audible displays in analytical chemistry; Mezrich, Frysinger and Slivjanovski (1984) explored auditory display techniques for multivariate date intended for oil well log data. Kramer (1991) used sound to display stock market data. Recently Blattner, Greenberg, and Kamegai (1992) enhanced the turbulence of fluids with sound, where audio was tied to the various aspects of fluid flow and vortices.

The work in auditory display has been greatly simplified by the existence of audio tools for the computer. *Auditory Displays*, edited by G. Kramer (1993), is the first book to bring together work in this rapidly evolving area. Algorithm animation, initially a beautiful but silent art, has burst into sound (Brown, 1992).

Cues and audio messages: Generally audio cues will be used as auditory icons or *earcons*, to provide information to the user. This information tends to be more abstract than that received through auditory displays. Auditory signals are detected more quickly than visual signals and produce an alerting or orienting effect (Wenzel, 1992). Nonspeech signals are used in warning systems and aircraft cockpits. Alarms and sirens fall into this category, but these have been used throughout history, long before the advent of electricity. Examples are military bugle calls, post-horns, church bells that pealed out time and the announcements of important events. Studies made by Buxton (1987) show that a significant amount of the appeal of video games is due to the audio.

Work on *auditory icons* was done by Gaver (1986) and *earcons* by Blattner, Sumikawa, and Greenberg (1989). Gaver uses sampled real-world sounds of objects hitting, breaking, and tearing as described in real-world sounds above. However, Gaver's auditory icons are meant to convey information of a more abstract nature, such as disk errors, etc. Gaver used the term "everyday listening" to explain our familiarity with the sounds of common objects around us. Blattner, Sumikawa, and Greenberg took musical fragments, called motives, and varied their musical parameters to obtain a variety of related sounds. We describe the construction of earcons below.

The Structure of Earcons

The building blocks for earcons are short sequence of tones called *motive*. From motives we can build larger units by varying musical parameters. The advantage of these constructions is that the musical parameters of rhythm, pitch, timbre, dynamics (loudness) and register can be easily manipulated. The motives can be combined, transformed, or inherited to form more complex structures. The motives and their compounded forms are called *earcons*, however, earcons can be any auditory message, such as real-world sounds, single notes, or sampled sounds of musical instruments.

A motive may be an earcon or it may be part of a compounded earcon. Let A and B be earcons that represent different messages. A and B, can be *combined* by juxtaposing A and B to form a third earcon AB. For example, A can be an earcon for "file" and B can be the earcon for "deleted," then AB is the earcon for "file deleted." Earcon A may be *transformed* into earcon B by a modification in the construction of A. For example, if A is an earcon, a new earcon can be formed by changing some parameter in A to obtain B, such as the pitch in one of its notes. A may be an earcon for "the computer is up" while the transformed earcon may inform the listener that "the computer is down." A family of earcons may have an *inherited* structure,

ERIC

where a family motive, A, is an unpitched rhythm of not more than five notes is used to define a family of messages. The family motive is elaborated with pitch (A+p = B) and then preceded by the family motive to form a new earcon, AB. Hence the earcon has two distinct components, an unpitched motive followed by a pitched motive with the same rhythm. A third earcon, ABC, can be constructed by adding a third motive, C, with both the pitch and rhythm of the second motive, but now has an easily recognizable timbre (A + p + t = B + t = C). For example, A could indicate a printer problem, and AB could indicate the printer is out of paper.

It is desirable to structure the meaning of inherited earcons hierarchically by the use of refinement to correspond to the hierarchical musical structure. The refinement process is used in semantic networks and object--oriented programming. The exact nature of the refinements is not given in the construction of earcons. This process of adding a motive that is the same as the preceding motive but with a variation in a musical parameter, can be used to form new earcons with a refinement in meaning.

An earcon is usually associated with an object. By an object we mean any identifiable structure, whether it is an abstraction such as an error message or a representation of something in the real-world. To display more than one earcon (or one object) we have to identify their temporal locations with respect to each other. Two primary methods are used: overlaying one earcon on top of another and the sequencing of earcons (Blattner, Papp, & Glinert, 1993). Some sort of merging or melding in to new sound could be considered, for example, the pitch of two notes can be combined into a third pitch, but we haven't attempted this.

Will sounds as abstract as earcons be accepted by the majority of users? The advantages are very clear: they are easily constructed on almost any type of workstation or personal computer. The sounds do not have to correspond to the objects they represent, so objects that either make no sound or an unpleasant sound still can be represented by earcons without further explanation. Data translated directly into sound requires less explanation or motivation than abstractions such as earcons. Auditory icons that make real-world sounds usually can be recognized quickly, however, most messages do not have appropriate iconic images. Several experiments have shown earcons are preferred over many other types of sonification and can be used successfully. Brewster, Wright, and Edwards (1993) found earcons to be an effective form of auditory communication. They recommended six basic changes in earcon form to make them more easily recognizable by users. These changes were: 1) use synthesized musical timbres, 2) pitch changes are most effective when used with rhythm changes, 3) changes in register should be several octaves, 4) rhythm changes must be as different as possible, 5) intensity levels must be kept close, and 5) successive earcons should have a gap between them.

Earcons are necessarily short because they must be learned and understood quickly. Earcons were designed to take advantage of chunking mechanisms and hierarchical structures that favor retention in human memory. Furthermore, they use recognition rather than recall. The tests run by Brewster, Wright, and Edwards had no training period associated with them. (They were heard only once before the test.) In spite of this, the subjects could use them effectively. If earcons are to be used by the majority of computer users, they must be learned and understood as quickly as possible taking advantage of all techniques that may help the user recognize them. One of these techniques is to introduce a new earcon as a sound-enhanced animated icon (Blattner, Papp, and Glinert, 1993).

The animated icons with earcons are implemented as *metawidgets* (Blattner, Glinert, Jorge & Ormsby, 1992; Glinert & Blattner, 1993). Metawidgets are autonomous objects, which can display themselves in a variety of ways including graphics and sound. The metawidget, or the meta-icon in this case, chooses its particular representation depending upon the other events on the screen and the state of the system. Metawidgets are defined by an object-oriented hierarchy and within each object there is the capacity for it to select methods that allow for alternative behaviors and displays.

Concluding Remarks

Educators have been limited by their tools. Although different types of media were available to display on the computer, it is only recently that we have been able to use interactive multiple media on computers. Recent results in psychology and ESM show that should be enticed into tasks by the display on the computer interface, then to maintain interest, we must continuously challenge and provide feedback for the student. Are multimedia user interfaces a new paradigm for our interaction with computer? Multimedia can teach us to play the piano, teleconference with foreign attendees while assisting us in language translation, teach languages with the sights and sounds of other cultures. Virtual reality is being used in medicine, architecture, and computer aided design. The addition of audio to the interface will provide much more mental engagement and emotional impact that we could have had before. We must take advantage of our new opportunities by using all the audio that our technology can provide and using it in new and creative ways.

References

Blattner, M.M., Sumikawa, D.A. and Greenberg, R.M. (1989): Earcons and Icons: Their Structure and Common Design Principles. *Human-Computer Interaction*, 4:11-44.

Blattner, Meera M. (1992) Messages, Models, and Media, *Multimedia Review*, Vol. 3, No. 3, pp 15-21.

Blattner, M.M. and Greenberg, R.M.(1992): Communicating and Learning through Nonspeech Audio, *Multimedia in Education* (A.D.N. Edwards & S. Holland, editors), Springer Verlag NATO ASI Series, pp 133-144.

Blattner, M.M., Greenberg, R.M., and Kamegai, M. (1992): Listening to turbulence: an example of scientific audiolization. In *Multimedia Interface Design*, edited by M. Blattner & R. Dannenberg. New York: ACM Press/Addison Wesley, pp. 87-104.

Blattner, Meera M., Glinert, Ephraim P., Jorge, Joaquim A., and Ormsby, Gary (1992): Metawidgets in the Multimodal Interface, *Proceedings of Compsac '92* Chicago, Illinois, September, pp. 115-120.

Blattner, Meera M., Albert L. Papp, and Ephraim P. Glinert (1993) Sonic Enhancement of Two-Dimensional Graphic Displays, In *Auditory Displays* (G. Kramer, editor), Santa Fe Institute Series, Addison-Wesley, Reading, Massachusetts.

Blattner, Meera M., and Glinert, Ephraim P. (1993) Programming the Multimodal Interface, To be published in the *Proceedings of ACM Multimedia '93*.

Brewster, S.A., Wright, P.C., and Edwards, A.D.N. (1993): An Evaluation of Earcons for use in Auditory Human-Computer Interfaces, To appear in the *Proceedings of INTERCHI '93 Conference on Human Factors in Computer Systems*, Amsterdam, The Netherlands, April.

Brown, Marc, and Hershberger, John (1992) : Animation of Geometric Algorithms: A Video Review, Digital Systems Research Center, Palo Alto, California.

Brown, M., Newsome, S. and Glinert, E.P.(1989): An experiment into the use of auditory cues to reduce visual workload. *Proceedings of the CHI'89 Conference on Human Factors in Computer Systems*. ACM Press, New York, pp.~339-346.

Buxton, William (1987) Private communication.

Csikszentmihalyi, Mihaly, and Judith LeFevre (1989) Optimal Experience in Work and Leisure, *Journal of Personality and Social Psychology*, Vol. 56, No. 5, pp 815-822.

Csikszentmihalyi, Mihaly (1988) The flow experience and its significance for human psychology, Csikszentmihalyi, Mihaly, and Isabella Csikszentmihalyi(Editors) *Optimal Experience*, Cambridge University Press, New York.

Gaver, W.W.(1986): Auditory icons --- using sound in computer interfaces. *Human--Computer Interaction*, 2(2):167-177.

Gaver, W.W. (1993): Synthesizing Auditory Icons, To appear in the *Proceedings of INTERCHI '93 Conference on Human Factors in Computer Systems*, Amsterdam, The Netherlands, April.

Gilman, Leonard, and Paperte, Frances (1952) Music as a psychotherapeutic agent, in *Music and Your Emotions.*, Liveright Publishing Corp. New York.

ERIC

Glinert, E.P., and Blattner, M.M. (1993) Programming the Multimodal Interface. Lawrence Livermore National Laboratory Tech Report UCRL-JC-112195 (preprint). Submitted for publication.

Herndon, Marcia and McLeod, Norma (1982) *Music as Culture*. Norwood Editions. Darby Pennsylvania.

Kramer, G. and Ellison, S. (1991): Audification--the use of sound to display multivariate data, In *Proceedings of the International Computer Music Conference*, Montreal, Canada.

Kubey, Robert, and Mihaly Csikszentmihalyi (1990) *Television and the Quality of Life: How Viewing Shapes Everyday Life.* Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Ludwig, L., Pincever, N. and Cohen, M. (1990): Extending the notion of a window system to audio. In *Computer*, 23:66-72.

Lunny, D. and Morrison, R.C. (1990): Auditory presentation of experimental data. In *Extracting Meaning from Complex Data: Processing, Display and Interaction,* edited by E.J.Farrell. Proceedings of SPIE, 1259:140-146.

Mansur, D.L., Blattner, M.M. and Joy, K.I. (1985): Soundgraphs --- a numerical data analysis method for the blind, *Proceedings: 18th Hawaii International Conference on System Sciences,* pp 163-174.

Mezrich, J.J., Frysinger, S. and Slivjanovski, R. (1984): Dynamic representation of multivariate time series data, *Journal of the American Statistical Association* ,79:34-40.

Miyoshi and Koizumi (1992) NTT's Research on Acoustics for Future Telecommunication Services, Applied Acoustics 36, pp 307-326.

Nettl, Bruno.(1983).*The Study of Ethnomusicology..* Univ. of Chicago Press, Urbana.

Rosenfeld, Anne H. (1985) Music, The Beautiful Disturber. Psychology Today, December, pp 48-56.

Rudnicky, Alexander II, and Hauptmann, Alexander G. (1992) Multimodal Interaction in Speech Systems, In *Multimedia Interface Design*, edited by M. Blattner & R. Dannenberg. New York: ACM Press/Addison Wesley, pp. 87-104.

Schafer, R. Murray.(1977).*The Tuning of the World.* Alfred Knopf, New York.

Skelly, Timothy C., (1991) The Design of Seductive Interfaces, ACM CHI '91 Tutorial (Conference on Human Factors in Computing Machinery), New Orleans, Louisiana, April..

Thomas, Tony (1973) *Music for the Movies,* A.S. Barnes and Company, South Brunswick and New York.

Weimer, David, and Ganapathy, S.K. (1992) Interaction Techniques Using Hand Tracking and Speech Recognition, In *Multimedia Interface Design*, edited by M. Blattner & R. Dannenberg. New York: ACM Press/Addison Wesley, pp. 87-104.

Wenzel, E.M. (1991): Three-dimensional virtual acoustic displays. In *Multimedia Interface Design*, edited by M. Blattner & R. Dannenberg, New York: ACM Press/Addison Wesley, pp. 257-288.

Yeung, E.S.(1980): Pattern recognition by audio representation of multivariate analytic data. *Analytic Chemistry*, 52(7):1120-1123.

Meera Blattner is also with The Department of Biomathematics, M.D. Anderson Cancer Research Hospital, University of Texas Medical Center, Houston, Texas 77030.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

ERIC

HYPERMEDIA AND COMMUNICATION: A CHALLENGE FOR INTERACTIVE LEARNING IN MATHEMATICS

ROSA MARIA BOTTINO(*), GIAMPAOLO CHIAPPINI(*), PIER LUIGI FERRARI(**)
(*) Consiglio Nazionale delle Ricerche - Istituto per la Matematica Applicata
(**) Università di Genova - Dipartimento di Matematica
Via L.B. Alberti 4 - 16132 Genova - ITALY

In this paper we describe a project which combines hypermedia with communication technologies in order to design and implement an interactive system (the ARI-LAB system) to assist pupils in arithmetic problem solving at the ages of 7 - 12.

In our project the opportunities supplied by hypermedia allow the users to build their own solutions to arithmetic problem by navigating through different computer-based environments which are integrated in the system; the environments include microworlds, databases containing sets of solved problems and a communication environment. The combination of hypermedia with communication facilities allows pupils to cooperate in order to solve problems, or even to interact with the teacher, through a local or remote network.

In the present work, technical and pedagogical aspects are closely connected. From one hand we base ourselves on research on hypermedia and communication systems and on the design and implementation of visual microworlds. On the other hand we take into account research in mathematics education with particular reference to the studies on applied problem solving, the role of visual representations as educational mediators and interactive learning.

The hypermedia system described in this paper is implemented in HyperCard 2.2 using HyperTalk programming language. It runs on Apple Macintosh computers with 13 inch video.

At present the ARI-LAB system is completely implemented. Its evaluation, which is not discussed here, will be carried out in 1993 in three different classes of primary school.

SOME GENERAL PERSPECTIVES ON THE LEARNING OF MATHEMATICS

In the last years problem solving has become one of the most important fields of research in mathematics learning and has been often regarded as a general paradigm of mathematical activity, also under the influence of constructivist theories. Among the large number of studies on problem solving performances we mention Buchanan (1987), Lemoyne & Tremblay (1986) and Lesh (1985). Other references are given through the paper.

The increasing importance of mathematics in everyday-life along with the diffusion of computers has induced a lot of educational projects to focus on strategy building abilities more than on written computational performance. This is due also to research that has shown a very poor correlation between the ability at performing written calculations and problem solving skills, as reported for example by Nesher (1986). An undue stress on computational skills has proved a serious obstacle to the development of problem solving processes.

Arithmetic competence, at the ages of 7-12, cannot be identified with the knowledge of a given set of concepts and rules, but rather with the ability at using arithmetic concepts and procedures in order to solve problems, in particular real problems. By 'real problems' we mean the problems pupils continually deal with in their experience and that are meaningful with regard to mathematical structure as well. Real problems often embody a lot of arithmetic questions which require pupils to deal with schematization processes that are much more complex than the ones involved in the stereotyped school problems designed only in order to test the application of a learned rule or definition. Research has pointed out the dangers of early formalization; the introduction of a formalism with no recognizable meaning or embodying a different meaning with respect to the problem situation (such as the formalism of subtraction for a difference problem) may lead pupils to the impossibility of building any strategy.

Therefore the task for an educational project should not be to teach the pupils a set of problem solving techniques. Pupils should be able to deal with a wide range of problem situations and construct specific strategies to get a correct answer, starting from their mathematical knowledge but also from their knowledge about the problem situation.

There is evidence suggesting that content-dependent strategies should be enhanced. For example, some studies, reported in Lesh (1985), show that good problem solvers use 'strong' content-dependent strategies, whereas average or poor solvers prefer 'weak' content independent ones, and that there is a good correlation between problem solving skills and the mastery of the specific problem situation.

Symbolic systems play a crucial role at different stages of the development of mathematical ideas and processes. Vygotskij (1962, 1978) and many others have stressed that languages (regarded as symbolic systems) may not only represent but also direct human thought. At this regard representations which are not completely arbitrary but preserve some analogical link with their related objects (such as geometrical figures, histograms, line of numbers, the language of coins) has proved very important. The work with visual representations differs from the manipulation of sets of objects just because of their status of symbolic systems (with a recognizable set of rules, and so on). Languages of this kind could act as mediators between the problem situation and its meanings and the mathematical concepts, relationships and processes involved, as widely discussed in Dreyfus (1991) and Barwise and Etchemendy (1991).

Social interaction (in the various forms it may be organized) has proved a powerful way to improve pupils' problem solving skills (see for example Doise et Al. (1975)). As a general reference on the fundamental role played by social interactions in shaping internal cognitive structures we refer again to Vygotskij (1978) and his notion of proximal development zone.

Outline of a learning system for arithmetic

The previous survey may explain what, in our opinion, should be purposes and features of a hypermedia system for arithmetic problem solving:

= the system should be oriented to the design of resolution strategies and not to the development of computational skills;

= it should allow pupils to build resolution procedures starting from their informal strategies and not to teach them "how to solve a problem";

= problems should be meaningful and not designed only in order to test the application of some rule;

= the system should promote the construction of content-dependent strategies and foster the development of arithmetic concepts and processes while keeping the pupils in touch with the meanings involved in the problem situation;

= it should supply a set of symbolic systems in order to represent a wide range of problem situations and, at the same time, to point out different mathematical structures and perspectives;

= it should allow actual interaction between groups of pupils; this means that pupils should be able to communicate to each other their tentative resolution procedures, including parts containing non verbal information, and to keep trace of the messages.

HYPERMEDIA ENVIRONMENTS FOR ARITHMETIC PROBLEM SOLVING

The educational purposes presented in the previous section have suggested us to design a system in which the technology of hypermedia is combined with distance and local network communication systems.

Our system can be regarded as an hypermedia system as far as a structured and connected set of environments of different nature is available to the user. Two different kinds of users are expected: the student who has to solve a given problem and the teacher who can configure the system according to the needs of her/his students.

The student interacts with two main environments: the *problem solving environment* and the *data-base environment*. The *problem solving environment* consists of three connected environments which are structured as follows:

= a strategy building environment where the user chooses an arithmetic problem out of a set prearranged by the teacher, and builds his/her solution strategy assembling written texts and visual representations built in the visual representation environment;

= a visual representation environment where the student can perform experiments through the interaction with a number of available microworlds;

= a communication environment where different kinds of interactions, between teacher and student and between student and student, are possible.

In the *data-base environment* the student can consult previously developed problems or can access the communication environment.

The *teacher environment*, available only to the teacher, allow him/her to configure the system according to the specific needs of the students involved.

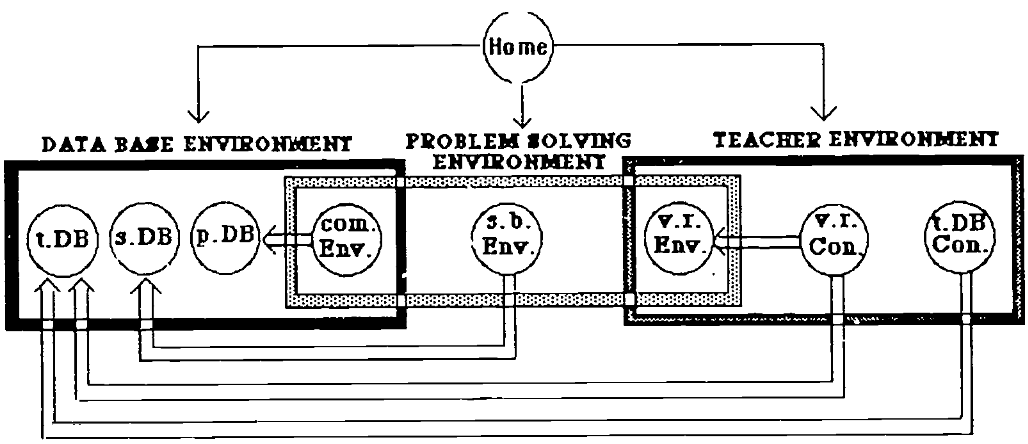
As a consequence of this structure, in our project hypermedia technology plays a twofold function: it is a tool for organizing working environments of different nature and for developing cognitive processes. In particular: as a tool of organization, it gives rise to a close interplay between various ways of action within different environments which are meaningful in relation with the field of knowledge involved; as a tool of thought, it may connect these opportunities of action to the activation of cognitive processes that allow the pupil to get the solution of a problem after the development of those planning, building, communicating and consulting skills that characterize the acquisition of 'doing arithmetic'.

In fig.1 the structure and the organization of the ARI-LAB system is sketched together with the main functional links among the environments the system is composed of.

The problem of browsing assumes a particular nature in our system. The goal (solution of an arithmetic problem) is clearly stated from the beginning. The user has to find out the opportunities offered by the system and to perform actions in order to fulfil the goal. In this sense, the search strategy he/she has to follow is an exploration strategy (see McAleese, 1989) coupled with the production of appropriate resolution actions. To perform this strategy a graphical interface seems appropriate.

The user interface is completely mouse-driven, and is designed to present the user with a sequence of structured display frames whose activation depends on his/her choice.

In the sequel we are going to give a brief account of the environments that are part of our system explaining our organization and implementation choices from both technical and educational viewpoints.



t.DB : teacher's Data-Base; **s.DB** : student's Data-Base; **p.DB** : collaborative partner's Data-Base; **Com.Env.** : Communication Environment; **s.b.Env.** : strategy building Environment; **v.r.Env.** : visual representation Environment; **v.r.Con.** : environment for the Configuration of the visual representation microworlds; **t.DB.Con.** : environment for the Configuration of the teacher Data-base.

⟹ : it represents the possibility of structural modifications of the environment indicated by the arrow as consequence of actions performed in another environment.

⟶ : it represents the links among stack home and the main environments of the system

Fig 1: General structure of the ARI-LAB system with the main links among its environments.

Problem solving environment

In the problem solving environment the user is presented with an arithmetic problem, expressed in written form, for which he/she has to find a solution, building a resolution strategy. This strategy is built in the strategy building environment. In order to build his/her strategy the student, at each step of the resolution process, can access the visual representation environment, choosing one of the available microworlds to get a representation of his/her resolution step. The complete solution for the given problem is the synergetic result of a group of actions and processes performed in the different microworlds. The user can activate the communication environment at any moment, in order to send, or receive, a message or the (complete or partial) solution of the current problem to, or from, some other user connected with him/her. The user can also save in the data-base the resolution he/she has received.

In fig.2 the user's interface of the system when the problem solving environment is activated is shown.

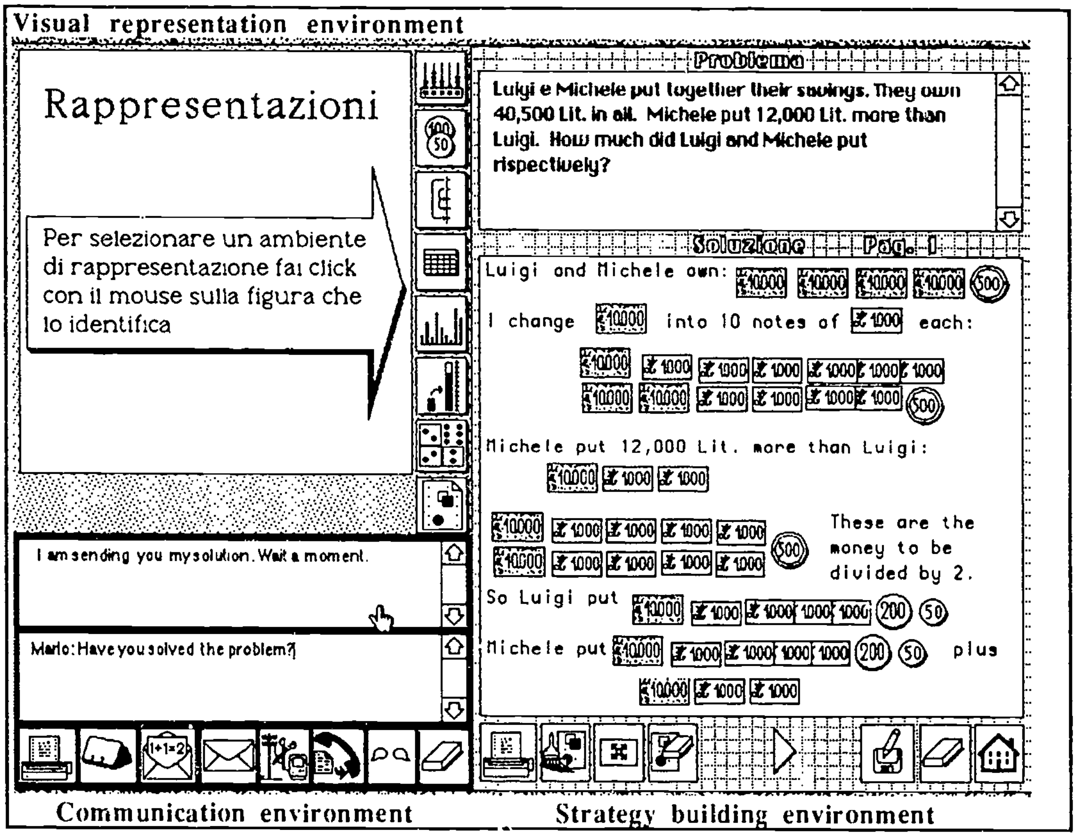


Fig.2: Interface of the prot.em solving environment with an example of a problem solution. This environment is composed of: a strategy building environmen', a visual representation environment, and a communication environment.

Visual representation environment

This environment is composed by a number of visual representation microworlds the user can interact with in order to build his/her own resolution strategy.

The representation microworlds available are at present: 'abacus', 'coins', 'simplified spreadsheet', 'line of time', 'histogram maker', two different microworlds for integer division, 'measurement division' and 'partitive division'.

Some microworlds are strongly characterized on the semantic ground ('coins', 'line of time'), others ('abacus', 'histogram' and 'spreadsheet') allow more formal, content-independent representations of mathematical objects, others ('integer division' microworlds) are of a mixed nature and have been planned based on observations of pupils' behaviours (see Boero et al., 1989).

In fig.3 the user's interfaces of the system when the microworlds 'line of time', 'coins' and 'simplified spreadsheets' have been respectively selected are shown.

We are not going to provide a detailed account of the features of the different microworlds here but we only point out the main technological and cognitive aspects that characterize them as a whole.

From the technological standpoint, this environment is built so that, at any moment, only one microworld can be selected by the user. Within each microworld there are links that point to a virtual set of options of graphic animations, which are of different kinds:

= there are animations consisting of the generation of symbolic objects both under the user's control (e.g. generating a particular coin, a ball in the abacus, an interval on the line of time, etc.) and automatically by the system, based on parameters given by the user (e.g. construction of an histogram given a table of data and a measurement unit);

= there are animations consisting of some transformation of symbolic objects (e.g. a coin may be changed and replaced by other coins). These actions are performed by the system after the user has determined their features, which are automatically checked by means of sets of rules incorporated in the system. For example, if the user propose an incorrect change for a coin, the system does not execute it but prompts a warning message.

= there are animations consisting of spatial movements of objects or of combined actions (generation, manipulation or movement of objects). For example, a coin may be moved and arranged (i.e. changed with other coins) in a different way.

In any microworld backtracking is always allowed in two ways, if some action does not give suitable results: step by step or automatically, according to the ways the action has been carried out.

From the cognitive standpoint, the microworlds offer the pupils the opportunity to perform resolution steps with an immediate visual feedback; in this way an interplay is promoted among anticipation, action and visual representation, which is crucial, in our opinion, to the development of arithmetic competence. Moreover, working within the microworlds strongly promotes the development of pupils' trial-and-error strategies, through a process of progressive schematization, where the statement and evaluation of hypotheses plays a crucial role (on the interplay between hypothetical reasoning and problem solving see Ferrari, 1992).

Pupils, when working in the microworlds, deal with symbolic objects, which in general can mediate between the semantics of the problem and the underlying mathematical structure; moreover, in some contexts, symbols can be used just in order to foster the construction of a mental representation of the problem situation.

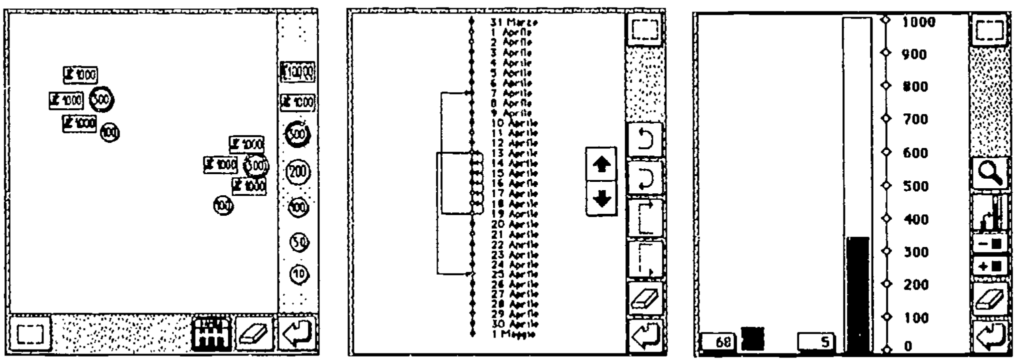


Fig. 3: Examples of the interfaces of three different microworlds: 'coins', 'line of time' and 'measurament division'. In each microworld various actions are possible in order to generate and to manipulate symbolic objects.

Communication environment

The communication environment is built in order to offer the user the possibility of communicating with another user who is in contact with him via a modem or a local network connection.

As previously pointed out, the communication opportunity is a key factor in the acquisition of problem solving skills, since it fosters the development of different ways of reasoning and of different resolution processes when dealing with a given problem situation. The system offers different communication opportunities to the user (distance communication via a modem connection or local communication with the teacher or with another student of his same class) which may induce different cognitive processes. The social interaction process mediated by the computer allows to:

= foster the construction of a stronger and more functional link between production and interpretation activity;
= aid the pupil to recognize and make clear his/her communication goal and to plan the related actions;
= use the different available languages (verbal, visual and mathematical) not only as tools to organize one's own thought but also to communicate the products of thought.

The analysis of these processes is quite new in educational computing research and seems very promising; we hope that the realized system can be an useful tool at this regard.

Our research focuses on exploring an alternative architecture to the one usually present in educational computing (student - computer) in which at least three agents take part in the learning process (student - computer - collaborative partner).

The user interface is realized so that the access to the communication environment is possible at any time of the work with the system. When an user wants to communicate with another user he/she can select this option with the mouse and, in this way, he/she accesses the communication environment with its own commands and links. When receiving a message, the user can decide to immediately read it or to delay the reading.

Data-base environment

The user access the data-base environment when he/she wants to consult previously developed and classified problems or when he/she wants to consult problems he/she had previously saved (his/her proper problems or problems received from other users). The classification of previously developed problems (by teacher or system designers) is done according to the visual representation form through which their solution is presented.

If there are two or more solutions for the same problem (using different visual representation forms) mutual links are present to associate them. In the user interface links are visualized so that the user can decide to consult alternative solutions by selecting the corresponding icon.

The user interface is built so that previously developed problems, problems solved by the actual user and problems received by other students are logically organized in different databases (see fig. 4). This is done in order to help students in their consultation activity. Student's own solutions, or solutions received from others students, are not classified according to the used representation form, as are the problems solved by the teacher, since in the process of building a solution, the student can decide to interact with more than one of the microworlds available, at the different stages of the resolution process.

The user can access the data-base environment at any time of his/her work within the system. In no case he/she can directly copy a part of a solution, taken from the data-base, in its strategy building area. On the contrary, he/she can send or receive problem solutions (or part of solutions) to other users connected.

Database exploration can start from the pupil's own experience (databases of the problems he/she has previously solved or received by another student, ordered according to time) or from the database of teacher's solved problems. Within this last database the student can perform the exploration according to two main criteria: he/she can look at other problems solved within the same microworld or follow the links leading him/her to solutions for the same problem within different microworlds. The semantics underlying the organization of the databases is very powerful: from one hand the relation to pupil's past experience gives the databases a temporal organization, according to the time problems have been solved or saved; from the other hand there is a semantics related to the way the different tools are used in the resolution of a given problem, which is completely different from a classification of problems by type.

Our choices are motivated by the following reasons:

= temporal organization, also because of students' age, is in our opinion the only criterium which can be fruitfully used by them in order to search for information in their past experience;
= database organization according to microworlds is due to the meaning of external representations as tools for structuring thought in relation to arithmetic knowledge.

ERIC

In this framework we think that the opportunity of solving the same problem within different representation microworlds and, at the same time, of using the same microworld in order to solve even semantically different problems may induce pupils to take into account the processes underlying the use of a tool in different contexts. This choice is related to the function of external representations in the resolution of problems, which can be regarded as tools of mind or, conversely, as the record of pieces of knowledge built elsewhere. Moreover, we think that the organization of teaching according to some taxonomy (such as addition, subtraction, ..., one step, two steps problems ...) only damage the construction of arithmetic knowledge, as widely reported in literature (Chevallard, 1989).

Teacher environment

As previously said, the system has an environment, available only to teachers, which allows them to set different layouts according to their choices. In particular the system allows the teacher to choose the microworlds accessible bythe students and to choose local network and distance communication connections. This implies that the interface of the system is dynamically and automatically changed in order to take into account the teacher's choices; moreover, in the data-base environment, links which associate different solutions for the same problem, are automatically updated. The teacher also uses this environment in order to produce problems solutions and to save them in the data-base.

DISCUSSION

The hypermedia system we have presented not only represents and structures information in a given field, but also provides a set of work . environments where the users can perform actions allowing them to find out their own resolution procedures. There is not only communication of knowledge but tools are provided to construct it by means of different ways of interaction.

This feature makes the system significantly different, from a conceptual viewpoint, from standard hypermedia educational systems, as far as the user becomes an active subject not only when navigating through the system, but also as builder of new knowledge.

The integration of hypermedia technologies with communication systems is another innovative characteristic which make the system an effective tool for collaborative work. Support for collaboration is very important because interactive learning and teaching are inherently collaborative and because hypermedia are, in principle, well suited for collaboration (see Tomek, 1991). Cooperation within two or more users requires study of intelligibility, which implies to consider problems related to the choice of the most easily understandable styles of hypermedia presentation. These problems were considered in this system, as for example in the ideation and design of visual representation microworlds, so that each representation of a situation (for example a given layout of the abacus or of an histogram) is completely understandable by itself (and so can be communicated by the user without the need of any further description).

A feature which put our system in a intermediate position between hypermedia systems and knowledge based systems (Kibby, 1991) is its characteristic of obtaining, under suitable conditions, an automatic feedback to a performed action based on system knowledge. Actually, in each microworld, the user's manipulation actions are controlled by a set of rules which prevent some specific incorrect step by the user. This characteristic can induce a reflection on given mathematics aspects (for example the decimal notation) even if the control of the strategic plan is up to the user.

The system has been designed based on the exploration of effective styles of presentation, with a shift of focus from technical bias to contents. Other innovative aspects are the capability of providing teachers and students with tools to develop learning according with consistent pedagogical models, and the chance given to the teacher to configure the system according with his/her needs.

The system offers the possibility of extending and of tailoring it to other mathematics areas, such as algebraic problem solving. This can be done quite easily since the system is done in such a way that it is possible to add other microworlds independently from the ones which are already present and to add examples of solved problems (and corresponding links) without having to intervene to the previous saved ones. The possibility offered to the teacher to configure the system according with his/her specific needs gives the opportunity to change the microworlds and the databases to work with without intervening in the general structure of the system.

ERIC

We think that the use of computers in this context may extend Vygotskij's idea of 'zone of proximal development', as the child, with the aid of computer technology, may do what she could not do by herself nor even with the assistance of an adult. This hypothesis, we are going to test in classroom experimentation, is based on the belief that the system can activate interactions highly effective in order to promote the development of arithmetic problem solving skills.

Within the different microworlds the work with the computer may take the form of a social interaction integrating action and communication in a dialectical way; the opportunity of learning also depends on the visual feedback available in the microworlds that allows the student to elaborate and justify action schemes oriented towards the attainment of a given goal. Moreover, the interaction between students mediated by the computer may affect the inner mental processes of the subjects and the nature of the communication between them as far as it is goal-oriented and always subject to rules.

At last, the navigation through the databases of solved problems leads to new forms of exploration and interaction with the knowledge they contain. The development of a resolution strategy can be performed only within the microworlds, which implies that a resolution by analogy with some other strategy included in the database has to be reconstructed through the set of commands of the microworld corresponding to the representation involved.

REFERENCES

Barwise, J., & Etchemendy J. (1991). Visual Information and Valid Reasoning, in Zimmerman. W. and Cunningham S. (eds.), *Visualization in Teaching and Learning Mathematics*, 9-24, Providence, RI (MAA Note Series, Vol.19).

Boero, P., Ferrari, P.L., & Ferrero E. (1989). Division Problems: Meanings and Procedures in the Transition to a Written Algorithm. *For the Learning of Mathematics* , 9 (3), 17-25.

Buchanan, N.K. (1987). Factors contributing to mathematical problem solving performance: an exploratory study. *Educational Studies in Mathematics*, 17, 97-123.

Chevallard Y.I. (1989). Arithmetique, algebre, modelization. Publications de l'IREM d'Aix-Marseille.

Conklin J. (1987). Hypertext - an Introduction and Survey. *IEEE Computer*, 20 (9), 17-41.

Doise, W., Mugny, G., & Perrret-Clermont, A. (1975). Social interaction and the development of cognitive operations. *European Journal of Social Psycology*, 5 (3), 367-383

Douady, R. (1986). Jeux de cadres et dialectique outil-objet. *Recherches en Didactique des Mathématiques*, 7 (2), 5-31.

Dreyfus, T. (1991). On the status of visual reasoning in mathematics and mathematics education. *Proceedings PME 15*, 1, 33-48.

Ferrari, P.L. (1992). Aspects of hypothetical reasoning in problem-solving. In J.P.Ponte, J.F.Matos, J.M.Matos and D.Fernandes (eds.), *Mathematical Problem Solving and New Information Technologies*, Nato ASI Series, 125-136.

Harel D. (1988). On Visual Formalism. *Communications of ACM*, 31 (5), 514-530.

Johnson-Laird, P.N. (1975). Models of reasoning. In Falmagne R.J. and Hillsdale N.J. (eds.), *Reasoning: representation and process in children and adults*, , Lawrence Erlbaum ass..

Kibby M. (1991). Intelligent Hypermedia for Learning. In Bottino, R.M., Forcheri, P. and M.T. Molfino (eds.), *Proceedings of the Sixth International PEG Conference*, 3-12.

Lemoyne, G., & Tremblay C. (1986). Addition and multiplication: problem solving and interpretation of relevant data. *Educational Studies in Mathematics*, 17, 97-123.

Lesh, R. (1985). Conceptual Analyses of Mathematical Ideas and Problem Solving Processes. *Proceedings PME 9*, 1, 73-96.

McAleese R. (1989). Navigation and Browsing in hypertext. In McAleese R. (ed.) *Hypertext: Theory into Practice*, Blackwell Publishing Corporation, Oxford, 6-44.

Nesher, P. (1986). Are mathematics understanding and algorithmic performance related?. *For the Learning of Mathematics*, 5 (3), 2-9.

Nielsen J. (1990). *Hypertext and Hypermedia*. Academic Press, San Diego.

Tomek I., Khan S., Muldner T., Nassar M., Novak G., & P.Proszynski (1991). Hypermedia-Introduction and Survey. *Journal of Microcomputer Applications*, 14, 63-103.

Vygotskij, L.S. (1962). *Thought and Language*. MIT Press, Boston.

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

ERIC

Intelligent retrieval of video stories in a social simulation

ROBIN BURKE
Institute for the Learning Sciences
Northwestern University, Evanston, IL 60626, USA

The stories of experts capture the realities of a realm of knowledge from an experienced perspective. Such stories are often important in novices' learning (Lave & Wenger, 1991), (Schank, 1991). In this paper, I describe SPIEL (Story Producer for InteractivE Learning), a multimedia retrieval system for educational stories, and its integration into an architecture for teaching complex social skills, such as management, diplomacy or selling.

SPIEL is embedded in an intelligent learning-by-doing architecture called Guided Social Simulation or GuSS. GuSS provides a social simulation in which students can practice social skills in an environment where mistakes are not as costly as the real world. The goal is to accomplish for the social environment what the flight simulator accomplishes for the physical environment of the cockpit (Haber, 1988). Currently, this architecture is being used to develop an application, YELLO, for teaching the fine points of selling Yellow Pages advertising. (For more on GuSS and its precursor ESS, see (Kass, et al., 1992), (Kass & Blevis, 1991) and (Blevis & Kass, 1991).)

Videotape, interactive videodisk and other multimedia technologies are frequently part of education in social skills, particularly in the corporate world (Craig, 1990). They are usually used to re-create social environments as in (Stevens, 1989), but this is not the role stories play in GuSS. In GuSS, the stories of experienced practitioners, preserved on video, help students reflect on their experience in the recreated social environment of the simulation, and let them compare their experiences with those of experts.

In the GuSS system, SPIEL is like an experienced instructor watching over the shoulder of the student. It monitors the simulation and presents stories from its library when they are relevant to the student's situation. Since stories describe principles as they apply in action, telling relevant stories in the context of a simulation is a particularly useful way to link the student's experience in the simulation to general principles of practice (Schank, 1991).

A YELLO example

For YELLO, we have gathered approximately 200 stories about selling Yellow Pages advertising from interviews with expert account executives. To see how the program combines simulation with the telling of these stories, consider the following representative scenario:

- The student is assigned to sell an advertising program to Swain Roofing Co. and is given the name of Mr. Ed Swain, the owner.
- The student calls the client and sets up an appointment.
- Upon arriving at the Swain's home, she meets Lucy Swain, Ed's wife and finds that Ed is not yet there.

- The student carries on small talk with Lucy but does not discover that Lucy "keeps the books" for the business and fails to gather any business information that Lucy could have provided.
- Ed Swain arrives.

This is a good time to give some feedback to the student. She has misjudged Lucy's role in the business and missed her opportunity to get some business information from an important source. Without feedback, the student might not realize that this opportunity existed. One of the stories in the YELLO library is relevant and the Storyteller calls the student's attention to it. Here is the text the student sees on the screen, along with a transcription of the story told in the video:

If you assume that Mrs. Swain will not have a role in the business of Swain Roofing, you may be surprised. Here is a story in which an account executive made a similar assumption and was wrong:

[Video: I went to this auto glass place one time where I had the biggest surprise. I walked in; it was a big, burly man; he talked about auto glass. So we were working on a display ad for him. It was kind of a rinky-dink shop and there was a TV playing and a lady there watching the TV. It was a soap opera in the afternoon. I talked to the man a lot but yet the woman seemed to be listening, she was asking a couple of questions. She talked about the soap opera a little bit and about the weather.

It turns out that after he and I worked on the ad, he gave it to her to approve. It turns out that after I brought it back to approve, she approved the actual dollar amount. He was there to tell me about the business, but his wife was there to hand over the check. So if I had ignored her or had not given her the time of day or the respect that she was deserved, I wouldn't have made that sale. It's important when you walk in, to really listen to everyone and to really pay attention to whatever is going on that you see.]

An assumption that a spouse will not participate in an advertising decision may be unrealistic. Think carefully about your assumptions.

This example illustrates the synergy between the simulation and the explicit instruction. Without the story to provide the impetus to examine the situation, the student might never realize what opportunities were missed. However, without active engagement in the simulation, the student might lack the motivation and context to understand and remember the story.

As seen here, SPIEL precedes each video clip with an introductory paragraph called a **bridge** and summarizes each with a **coda** paragraph. These text items must be generated by the program at retrieval time. The bridge explicitly connects the story context to the student's activity; the coda brings the student back to the events of the simulation by suggesting possible actions.

Teaching with stories

This example shows how a storyteller can teach by telling the right story at the right time. Intelligent tutoring systems have traditionally used modeling techniques to determine when to intervene and what to say, but they have been applied where the knowledge being taught is fairly narrow and well-defined. A system to teach social skills cannot operate in this way. No one model of expert social knowledge exists: there may be as many "right" ways to solve a problem as there are experts. To teach social skills, we cannot rely on a single body of knowledge, representing what students are to learn. We can, however, gather and use local knowledge (Geertz, 1983), such as stories told by practitioners.

The student's own social knowledge is large and idiosyncratic. It would be very difficult to gather enough data to build a model of what each student knows. However, SPIEL does not need to model the state of the student's social knowledge in detail because it is not seeking to address specific errors with its stories. The storyteller's goal is to expand the student's vision of the domain beyond what is happening in the simulation at any given time by bringing in relevant experiences that others have had. This may involve student errors or not, and there may be stories with conflicting messages. In apprenticeship situations, stories are used to show possibilities outside of the current experience in exactly this way (Lave and Wenger, 1991). Consider the following example:

- The student attempts to get the customer to buy a larger ad by demonstrating that co-operative programs with manufacturers will pay for part of the increase.
- The customer sees this as a way to pay less for the same ad, and decides not to increase the ad's size.
- The student fails to sell the larger ad.

The student has not made an error in introducing the idea of co-operative advertising. Co-op programs are frequently used with great success, and the storyteller has stories about how they have been invoked to help sell an ad program by making it more affordable. It is important to tell such a story at this point, because it shows the student the difference between the real world and the simulation. The student will be interested in hearing such a story, since it confirms her chosen approach.

Learning by doing is frequently driven by failures such as this: unsuccessful results or unmet expectations. Failure-driven learning theory in cognitive science (Schank, 1982) holds that there is a central cycle in performance learning. Learners make predictions about results of their own actions or the behavior of others, and test these predictions by observing what happens. If all goes well, these expectations feed into decisions about the next action or prediction. However, if the expectations fail, the learner has failed to understand something about the world. To prevent future failures, the learner must explain how the failure occurred in sufficient detail to allow the problem to be fixed.

Explanation is often the most difficult aspect of the prediction-observation-failure-explanation cycle. The learner has to identify why the prediction was wrong and what to do to prevent future errors. Stories can help this kind of learning in two ways. If the student has failed in some task, the storyteller can recall a story that helps the student explain the failure, as the "co-op" example showed. If the student is heading in the wrong direction without being aware of it, a story can point out the problem by showing where the direction is leading. It can spur learning by getting the student to recognize a failure. This is the role of the "Wife watching TV" story.

The failure-based learning model indicates that what is important to model about students is their state of interest, not their state of knowledge (Edelson, in preparation). If the system can identify what the student is interested in explaining, it may be able to respond with a relevant story. SPIEL has nine storytelling strategies, based on this model of learning from experience. The first example showed the use of the **Warn about assumption** storytelling strategy. This strategy calls upon the storyteller to tell a story about an incorrect assumption when it has evidence that the student may have a similar assumption, thereby getting the student to see the inappropriate assumption as a failure. The "co-op" example cited above uses the **Demonstrate opportunities** strategy, which calls for the use of a story about a successful plan when the student has failed while executing a similar plan, helping the student explain the failure and compare the situation in the simulated world with the world of practice.

Using storytelling strategies

Stories themselves are the source of SPIEL's knowledge about what to teach. If an expert has considered an issue significant enough to recall and tell a story about it, the system should try to

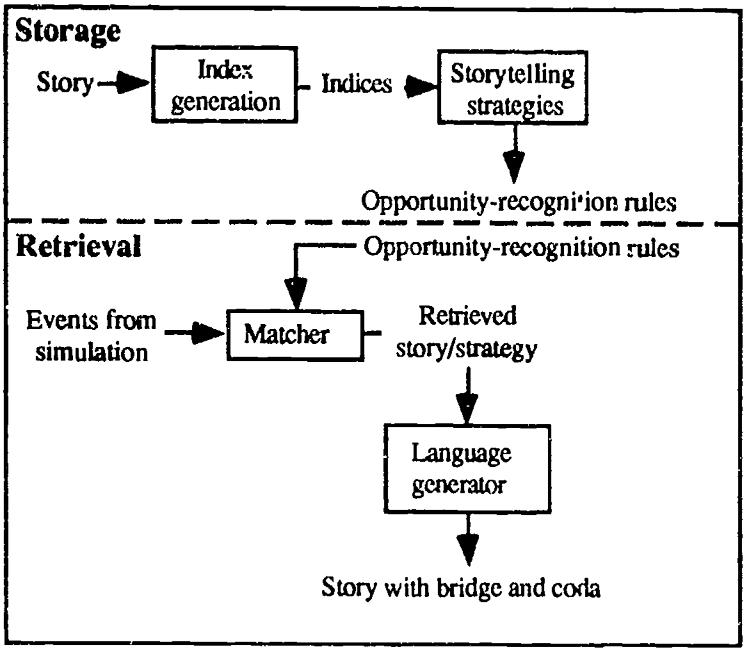


Figure 1. SPIEL's processing

find an opportunity to tell that story. SPIEL finds tutorial opportunities for the use of these stories by considering each story it knows about in the light of each of its storytelling strategies. To recognize these opportunities as they arise in the GuSS simulation, SPIEL creates a set of opportunity-recognition rules that can be applied while the student is interacting with the simulation. This architecture divides the system's processing into storage-time operations and minimizes the amount of inference that need be performed during student interaction. The steps of processing can be summarized as follows (see also Figure 1).

At storage time:

1. Indices are attached to each story in the database.
2. Each of SPIEL's storytelling strategies examines each story index. If the strategy is applicable, it will generate a set of opportunity-recognition rules for the story.

At retrieval time:

1. Opportunity-recognition rules are matched against the state of the simulation.
2. When a story is successfully retrieved, natural language texts (the bridge and coda) are generated that integrate the story into the student's current context.

In the following sections, I will describe SPIEL's indexing method and the storytelling strategies. A discussion of the system of opportunity-recognition rules requires an understanding of the internal operation of the GuSS simulation and is beyond the scope of this paper. Those aspects of SPIEL and GuSS are covered in (Kass, et al., 1992) and (Burke, in preparation).

Indices

SPIEL's indices are created manually. Since YELLO's stories are in video form, automatic processing would entail speech (and possibly gesture) recognition as well as natural language understanding. SPIEL's design therefore calls for a human indexer to read a story or watch it

Setting:	
	step-of-sales-process(salesperson, pre-call)
	business-type(service)
	business-size(small)
	business-partnership(auto-glass man, wife)
	married(auto-glass man, wife)
	sales-target(salesperson, auto-glass company)

Anomaly:				
	salesperson	assumed	wife	theme

	Assumed	Actual
Theme:	housewife	business partner
Goal:	hospitality	help in decision
Plan:	small talk	evaluate presentation
Result:		positive evaluation
Pos. side-effect:		sale for salesperson
Neg. side-effect:		

Figure 2. Index for "Wife watching TV."

being told, and use an indexing tool to compose indices that capture interpretations of the story's meaning. One such interpretation of the "Wife watching TV" story would be "The salesperson assumed the wife would have the role of housewife, but actually she was a business partner."

This general form, "X believed Y, but actually Z," is a form of **anomaly**. An anomaly is a failure of expectation that requires explanation (Schank, 1982). Typically, anomalous occurrences are what make stories interesting and useful, and they are a natural way to summarize what a story is about. Anomalies are especially important in stories about engaging in social activity since students are learning what expectations they should have and how to address expectation failures.

The anomaly forms the core of the index. Figure 2 shows the whole index for "Wife watching TV." It contains, in addition to the anomaly,

- the setting, the story's position within the overall social task, including a representation of the social relationships between the actors in the story, and
- intentional chains surrounding and explaining the anomalous occurences.

(This index structure is derived from the Universal Indexing Frame (Schank, et al., 1990). See also (Domeshek, 1992) .)

This structure is considerably more complex than indices that are typically found in information retrieval or even in the field of case-based reasoning from which this framework derives. There are several reasons for this. Information retrieval systems that store text can search their textual contents, if necessary. Since there is no computer-interpretable text in a video clip, all of the information needed for retrieval must appear in the index. SPIEL's indices are detailed because the system needs to find detailed correspondences between the events in the simulation and the stories it tells. Its stories must be closely relevant, so there must be enough detail in the index to verify relevance.

Storytelling strategies

A tutorial storyteller needs indices to know what its stories are about. It also needs to know how stories can be usefully employed. For maximum educational impact, a story must appear precisely at the time that the student is prepared to learn from it. A storytelling strategy, such as

ERIC

Strategy	Story is about:	Tell story when:	Explanation
Demonstrate opportunities	the positive result of a particular course of action.	the student has executed a similar course of action with poor result.	The student may conclude the plan is a bad one.
Demonstrate risks	the negative result of a particular course of action.	the student has executed a similar course of action with good result.	The student may conclude the plan is a good one.
Demonstrate alternative plan	a successful plan to achieve a particular goal.	the student executed a very different plan and failed to achieve goal.	The student may not know about other ways to achieve the goal.
Warn about hopes	a desire that someone had that was not realized	the student appears to have the same hopes.	The student may need to know what is reasonable to hope for.
Warn about fears	a fear that someone had that did not materialize.	the student appears to have the same fear.	The student may need to know what fears are realistic.
Warn about assumption	an assumption that someone made that did not hold.	the student appears to have made the same assumption.	The student may need to recognize and evaluate assumptions.
Reinforce plan	a successful plan.	the student has started to execute a similar plan, but won't see the result for awhile.	The student may interpret lack of positive response as a bad sign.
Warn about plan	an unsuccessful plan.	the student has started to execute a similar plan, but won't see the result for awhile.	The student may interpret lack of negative response as a good sign.
Explain other's plan	a plan that the student might not know about.	student has just observed someone execute a similar plan.	The story explains the un familiar plan.

Figure 3. Storytelling strategies

Warn about assumption combines a characterization of a type of story: a story about an assumption that turned out not to hold, with a method for recognizing opportunities to tell such a story: look for evidence that the student may have a similar assumption. Each strategy has a pedagogical basis: the **Warn about assumption** strategy gets the student to recognize the possibility of an erroneous assumption, a necessary pre-condition to learning how to avoid such assumptions. Figu.e 3 explains SPIEL's nine storytelling strategies by showing what stories they apply to, what storytelling conditions they impose, and, briefly, their rationale.

One interesting consequence of SPIEL's storytelling strategies is that the system must retrieve stories based on structured comparisons between stories and situations. It differs from both information retrieval and case retrieval systems that use overall similarity between retrieval cue and retrieved item as the basis for recall. Retrieval of pedagogical stories requires careful assessment of similarities, dissimilarites and other relations between cue and story.

SPIEL examines all indices for each story using every storytelling strategy. Each compatible combination of story index and strategy yields a recognition condition description (RCD), a representation of a tutorial opportunity, a situation that could arise in the simulation that would make the story worth telling. For example, this is the recognition condition description for telling the "Wife watching TV" story using the **Warn about assumption** strategy:

WHEN the student is in the pre-call (information gathering) stage and speaking to someone who is a partner in the business and the spouse of the decision maker,

LOOK FOR That person having important information to provide, and
The student having failed to gather than information by the end of the conversation,

THEN TELL "Wife watching TV" AS a "Warn about assumption" story.

The RCDs feed into the rule generator that translates them into rules for recognizing the conditions they describe within the GuSS system. Creating the RCD requires knowledge about the kinds of conditions that are present in the simulation, such as what role the student plays and the kinds of actions available. Each strategy selects different aspects of an index as important and performs operations to predict how the story could be told with that strategy. Some strategies require knowledge about how the student's mental state might give rise to certain actions. Students do not always act in ways that clearly indicate their beliefs, but if they do, the system should be prepared to respond. In this example, the **Warn about assumption** strategy has to make use of the knowledge that the student's failure to ask useful questions is a good indicator of an assumption on the student's part that there is no information to gather.

Conclusion

Through SPIEL, students can reference a large body of expert knowledge, as preserved in the stories experienced practitioners tell. They do so without having to master a hypermedia interface or formulate retrieval requests. SPIEL watches student activity and recalls stories that address the student's interests of the moment. By modeling student interest using a failure-based learning model, SPIEL can tell relevant stories without a detailed representation of social reasoning. Currently, the system has 200 stories about selling Yellow Pages advertising, indexed by structured indices that capture interesting or anomalous aspects of the experts' narratives.

SPIEL uses knowledge-intensive methods to turn its general indices into specific retrieval rules appropriate to a particular application. This means that stories can be indexed without necessarily knowing the exact circumstances in which they might be told. SPIEL has a taxonomy of storytelling strategies that use similarity, dissimilarity, and other relations to make structured, strategic comparisons between stories and the situations in which they are retrieved.

References

Blevis, Eli, & Alex Kass (1991). Teaching by Means of Social Simulation. In *Proceedings of the International Conference on the Learning Sciences*. Northwestern University: AACE.

Burke, Robin (1992). Knowledge acquisition and education: a case for stories. In *Proceedings of the Symposium on Cognitive Aspects of Knowledge Acquisition*. Stanford University: AAAI.

Burke, Robin (in preparation) *Representation, Storage and Retrieval of Stories in a Guided Social Simulation*. PhD thesis, Northwestern University.

Burke, Robin, & Alex Kass (1992). Integrating Case Presentation with Simulation-based Learning-by-doing. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Assoc.

Craig, Robert L., editor (1990) *Sales training handbook: a guide to developing sales performance*. Englewood Cliffs, NJ : Prentice Hall.

Domeshek, Eric A. (1992) *Do the Right Thing: A Component Theory for Indexing Stories as Social Advice*. PhD thesis, Yale University. Issued as Technical Report No. 26, Institute for the Learning Sciences.

Edelson, Daniel (in preparation) *Learning from Stories: Indexing, Reminding and Questioning in a Case-based Teaching System*. PhD Thesis, Northwestern University.

Geertz, Clifford (1983) *Local knowledge: further essays in interpretive anthropology*. New York: Basic Books.

Haber, Ralph Norman (1988), Flight Simulation. *Scientific American: Trends in Computing.*

Hammond, Kristian J. (1986) *Case-based Planning: An Integrated Theory of Planning, Learning and Memory*. PhD thesis, Yale University.

Kass, Alex, & Eli Blevis (1991). Learning Through Experience: An Intelligent Learning-by-Doing Environment for Business Consultants. In *Proceedings of the Conference on Intelligent Computer-Aided Training*. Houston: NASA.

Kass, Alex, Robin Burke, Eli Blevis, & Mary Williamson (1992). *The GuSS Project: Integrating Instruction and Practice through Guided Social Simulation* (Technical Report No. 34). Institute for the Learning Sciences.

Kolodner, Janet L., & Menachem Y. Jona (1991). *Case-based Reasoning: An Overview* (Technical Report No. 15). Institute for the Learning Sciences.

Lave, Jean, & Etienne Wenger (1991). *Situated Learning: legitimate peripheral participation*. Cambridge University Press.

Salton, Gerard, & Michael J. McGill (1983). *Introduction to modern information retrieval*. New York: McGraw-Hill.

Schank, Roger C. (1982). *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press.

Schank, Roger C., et al. (1990). *A Content Theory of Memory Indexing* (Technical Report No. 2). Institute for the Learning Sciences.

Schank, Roger C. (1991). *Tell Me a Story*. Charles Scribner's Sons.

Stevens, Scott M. (1989). Intelligent Interactive Video Simulation of a Code Inspection. *Communications of the ACM*, 32(7), 832-843.

Williams, Susan M. (1991). *Putting Case-Based Instruction into Context: Examples from Legal Business, and Medical Education* (Technical Report). Learning Technology Center, Vanderbilt University.

Funding

This work is 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 ONR under contract N00014-J-1987, and by the AFOSR 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, an Institute partner, and from IBM.

Acknowledgments

The GuSS project has prospered under the guidance of Roger Schank, Alex Kass and Eli Blevis. I would like to acknowledge their contribution to the ideas in this paper. GuSS is the product of many minds and efforts including those of Mark Ashba, Greg Downey, Debra Jenkins, Smadar Kedar, Jarret Knyal, Maria Leone, Charles Lewis, John Miller, Thomas Murray, Michelle Saunders, Jeannemarie Sierant, and Mary Williamson. The sales and sales training organizations of Ameritech Publishing have also contributed to this research in many ways.

A Multimedium Social Learning System[1]

Tak-Wai Chan

Institute of Computer Science and Electronic Engineering
National Central University
Chung-Li, TAIWAN 32054
R. O. C.
email: chan@sparc15.src.ncu.edu.tw
fax: 886-3-4255830

Introduction

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 environment. By a multi-channel system, we mean a system where a student engages in a social learning environment and interacts with various agents who may be human beings or computer simulated agents. Furthermore, while multiple learning cues from different media can be received in multimedium platforms, artificial intelligence techniques are adopted to search for adaptability and flexibility for both individual and group learning. Thus, *multi-channel = social learning + multimedia + artificial intelligence.* This paper reports some issues arisen in conducting the research. In next section, the evolution of the notion of social learning systems is traced. Then we describe research perspectives of LISA, followed by a brief discussion in the last section.

Background

There is an old Chinese adage stating that the *Five Educational Objectives* are *Morals*, *Knowledge*, *Athletics*, *Sociability*, and *Art*. Among these five educational goals, knowledge has become so important that whether one can financially survive in a modern society depends on what and how much one can learn. To meet the demand of the society, formal schooling has become a prevalent method of education. Unfortunately, school teachers are heavily burdened by their role of transferring knowledge to students — so heavily that they rarely find enough time to pay attention to the students' individual needs. Thus, schools can only function as factories of producing massive graduates (Figure 1).

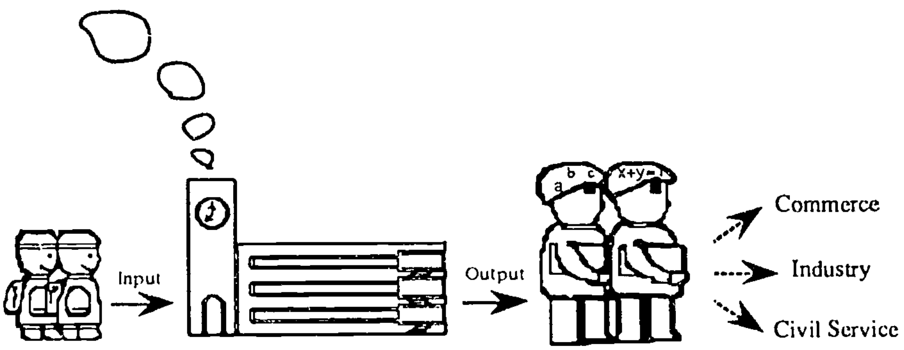


Figure 1 Schools as Factories

1 LISA is currently funded by National Science Council, Taiwan. Two dozen of researchers and graduate students from different universities of Taiwan are involved in the project.

111

Computer has already proved its momentous role in engineering in the Seventies and in business in the Eighties. The emergent multimedium technologies, on the other hand, reinforce the vision that computer can bear the same importance in education by the turn of this century. At the early stage of the development of computer aided instruction (CAI), many started with stimulus-response approach because of the influence of the behaviorism in psychology and realizing that computer can provide an interactive learning environment. Today, many CAI systems have shown their strong abilities in various areas in education. Starting from the Seventies and being spurred on by the interest of artificial intelligence, intelligent tutoring systems (ITS), drew attention of many computer scientists and cognitive psychologists. ITS, modeled after the idea of a one-on-one private tutoring, promised to pay better attention to individual student needs. Thus, a substantial amount of ITS research in the Eighties was devoted to the study of student modelling and cognitive diagnosis as well as representing domain knowledge and teaching strategies.

Since the mid of the Eighties, the role of the computer as an authorized teacher for transmitting knowledge has been being challenged by some researchers. Self and his colleagues suggested that the computer can be treated as a collaborator (or a co-learner) [Self 85, Self 86, GiSe 88, CuSe 90] and argued that the teacher's role played by the computer should be de-emphasized. Chan and Baskin [ChBa 88, ChBa 90] proposed that the computer can be simulated as two coexisting agents, a teacher and a learning companion. The addition of the learning companion to the traditional ITS model yields collaboration as well as competition. More recently, Pathepu, Greer, and McCalla [PaGM 91] suggest an inverted ITS model that treats the computer as a student and the human learner as a teacher. In summary, whether it is a collaborative system such as People Power [DiSe 92], a learning companion system (LCS) such as Integration-Kid [Chan 91], or a learning by teaching system, the scenario has the following implications:

- artificial learner can be modeled by the computer;
- human students make cognitive gains by collaboration, competition, or teaching the artificial learner; and
- the computer can model multiple agents with different roles, like LCS, forming a small society in learning.

The last implication is to recover the social context that has been lost in most computer-based learning environments. Despite the existence of an artificial learner in the system, this line of research coincides with the current interest of distant learning or computer supported collaborative learning by way of computer network. Moreover, the advancement of multimedium platforms provides powerful devices to support complicated peer dialogue, gestures, and so on, making complex social learning systems more viable than before.

Psychologists have conducted various studies on cognitive development under social interactions. Piaget [Piag 67] regarded peer interaction as an ideal forum for helping children shift away from egocentric views, enable them to consider multiple perspectives, foster a more comprehensive mature conception to emerge, and thus take a leap to a higher level of understanding. Vygosky [Vygo 78], the developmental theorist who most emphasized the social nature of cognition, argued that cognitive development process is the gradual internalization and personalization of what was originally a social activity. Learning in a social environment, thus, is largely the individual re-enactment of the cognitive processes that was originally experienced in society.

Research Perspectives

We conceive the LISA project at the beginning of 1992, but its path can be traced back to 1988. Thus, we describe the project in a few stages. In the first stage (1988 to 1992), after the introduction of the LCS idea [ChBa 88, 90], the first prototype of LCS, Integration-Kid, in the domain of integration was implemented [Chan 89, 91]. A knowledge-based architecture, Curriculum-Tree, for LCS has also been developed [Chan 89, 92]. In 1991, instead of adopting a computer simulated learning companion, we implement a distributed LCS, Distributed WEST, to study learning effectiveness and motivation when human students learn together via network [Ch+ 92]. The system is a revised version of a CAI classic WEST developed for the PLATO Elementary Mathematics Project and ITS program [BuBr 76]. Later, the design of a set of small pedagogical programming languages was completed and used in the course "Introduction to Computer Science" for freshmen [ChWa 92]. Starting from 1992 where the second stage (1992 to 1995) began, more researchers joined the project and the research perspectives have been widened and the level of study deepened. So far, in that stage, we have developed a model of learning stages, OCTR, as a conceptual design framework for LISA [CLLK 93]. Currently, several sub-projects are launched out into further exploring social learning environments, extending the Curriculum-Tree knowledge-based architecture, investigating student model and teaching strategies, inquiring relationship between meta-cognition and social learning, and experimenting and evaluating implemented prototypes with human subjects. Finally, in the last stage (1994 to 1996), apart from continuing to inquire the fundamentals of social learning, we plan to integrate technologies developed and evaluate the system in a large scale since by then the courseware for a class of freshmen will be finished.

Domain

We have designed a coherent set of highly simplified programming languages as the teaching material, which consists of six languages, Core, Block, Lisp, ADT, Prototype, and SOOP [ChWa 92]. Core is designed for beginners to learn programming. It comprises of a few basic elements of most high level programming languages. Block can be viewed as a subset of Pascal by teaching block structure via nested function declarations as well as procedural abstraction, structured programming, and top-down design. Lisp is a tiny subset of Common Lisp where we teach recursion and a large data type, list. The last three languages are designed for teaching the prevalent object-oriented programming. ADT imitates CLU's Abstract Data Type [LiAS 77]. Prototype is a simplified language described in [Lieb 86] for introducing the concepts of prototypical objects, delegation, and message-passing. Finally, SOOP is a Small Object-Oriented Programming language that is a combination of ADT and Prototype.

OCTR: A Conceptual Design Framework

Social learning might not be ample for the students' whole learning process of a subject matter. In seeking a "complete" conceptual design framework, we resorted to the evolving learning stages of students and proposed a model, OCTR [CLLK 93], that is an extension of apprenticeship learning [CoBN 89] and the three modes of learning by Rumelhart and Norman [RuNo 78]. OCTR consists of four stages : (i) Orientation (prior knowledge connection), where the system helps the students relate their prior knowledge of the domain to learn; (ii) Coaching (knowledge growth), where the teacher first models the task for the students, then help them handle the task on their own mainly by scaffolding and fading; (iii) Tuning (knowledge articulation), where knowledge restructuring is exercised via peer interaction and the teacher plays a less active role; and (iv) Routinization (knowledge solidification), where students solidify their knowledge by keeping on practicing, perhaps under some form of peer pressure.

Learning under OCTR advances gradually as the amount of social activities increases. Interestingly, OCTR can also serve as a framework for describing the cognitive development and motivation of students, categorizing many existing teaching strategies, and better understanding how to introduce meta-cognition and attitude in learning systems. Perhaps, most important of all, OCTR can bridge traditional CAI, ITS, and social learning systems. Traditional CAI seems to be best on orientation and routinization while ITS concentrates more on one-on-one coaching (because there are more intriguing artificial intelligence issues such as student modelling in the coaching stage than orientation and routinization stages); social learning, on the other hand, is more appropriate in tuning and routinization stages.

Finally, we may use OCTR to describe the different educational practices in the East and the West. In the competitive educational systems developed in some Asian countries, the survival of public examinations is the main objective of almost all students, the emphasis is mainly on modelling (a part of coaching) and routinization, and leaving other parts of coaching and tuning unattained. However, many Asian families employ private tutors for their children. In the West, orientation and tuning seem to be the key parts of the classroom teaching. Orientation and tuning can encourage independent thinking and innovative ideas; but inadequate emphasis on routinization may result in the decline of the students' examination performance. This difference in educational philosophies partly reflects the cultural difference between the West and the East.

Learning Environments

LISA is a system composing a number of sub-systems of different social learning environments. Currently, we are investigating some learning environments in the large spectrum between two simple models: the centralized LCS model such as Integration-Kid where there is only one artificial learning companion and the distributed LCS model such as Distributed WEST where there is only one human learning companion. For the centralized LCS's, we are exploring three models: Three's Company, GLASS, and LNET. Three's Company (Figure 2a) is a direct extension of the original LCS where the user learns together with two computer simulated learning companions. The different performance and persona of the *two* companions generate interesting cognitive and motivational issues. GLASS (Glassroom society dialogue model) (Figure 2b), is a more complex model. It consists of two parts: the inner world (glassroom) and the outer world. The outer world is an LCS model. The inner world is the subject to be observed and discussed by the outer world. The glassroom is a general term. It may be an object, an episode of video, or another small learning society. A daily life example of this model perhaps is the operating room in a hospital where medical students outside the room observe the operation process through a glass window and are explained by an instructor some critical points in the process. LNET stands for Learning by NEgotiated Teaching (Figure 2c) where the human student negotiates with the teacher a mutually acceptable learning goal and learns after the teacher before teaching the computer student; and the process is repeated.

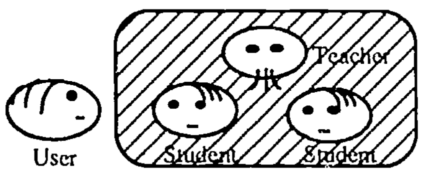


Figure 2a Three's Company

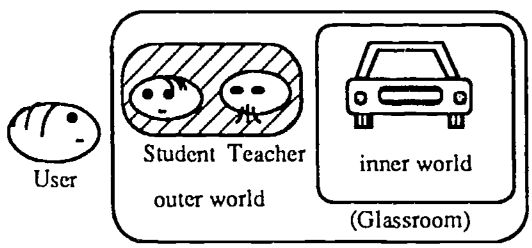


Figure 2b GLASS

The investigation of distributed social learning models (Figure 2d) was led by the anticipation of the integration of ISDN technology and satellite communication and that there will be more schools not located at centralized places in future. For example, the *synchronized* model is one where all students face the same computer teacher. If the teacher asks all students the same question, students will then communicate with their own computer companion before giving an answer. If the teacher does not offer comments to any individual student, then each student may have a private computer teaching assistant to help. Also, it shows statistics of the students' answers and selects excellent answers or common mistakes to discuss. Because all students learn from the same teacher, it needs to exert some synchronized mechanism during the learning process. Another model we are looking at is the distributed version of reciprocal teaching [PaBr 84] where an implicit cognitive task is partitioned into several explicit sub-tasks taken by different students, and they take turns to undertake each of these sub-tasks. This is indeed a form of distributed role playing game but in the context of learning. Besides, we are using distributed social learning systems as intermediate systems before accomplishing a centralized learning system. For example, to design LNET, a centralized learning by negotiated teaching system, we start with some distributed versions of the system first because a distributed version would be simpler and easier to understand. Finally, we are also developing a framework to classify and analyze distributed social learning systems.

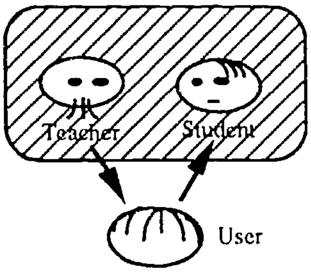


Figure 2c Learning by Negotiated Teaching

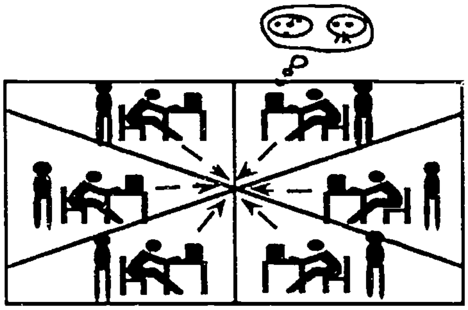


Figure 2d Distributed Social Learning Systems

Student Modelling and Teaching Strategies

Because of better focus on learning environments, we do not include student model in the current prototypes such as Three's Company and GLASS. Student modelling problem is more complex in social learning systems than in ITS since a set of student models is required, instead of one. Apart from cognition, student modelling embraces issues of meta-cognition, motivation, and attitude. Several sub-projects set forth in this direction. Since we can view learning in social environment as successive interpretations of the subject by different agents, a sub-project begins to seek for a representation of conversational model of agents. Another research attempts to raise motivation by developing effective dialogue model among agents. Also, engendering meta-cognition through social learning is another sub-project's focus. To support teaching strategies in a social learning environment, we also conduct a research on examining how to index examples.

Besides these issues, social learning represents a wide range of opportunities for contribution of students' attitude and sociability. For example, in social learning environments, students learn that making mistakes is a natural phenomenon in learning and experience sharing of success in collaboration. In reciprocal learning, if there is a critic, students will find that negative opinions are constructive to the outcome, especially when the outcome is compared with others. In learning by teaching, the user helps computer learn, instead of being helped. This change of perception of interaction and the experience of playing an adult role in a productive way may help students develop a sense of personal adequacy.

Multi-Agent Interface Design

Apart from text and voice, different agents' graphical facial expressions and gesture can potentially animate social dialogue and emit informative elements. It remains to be a research issue to generate learning information with higher density by coordinating these multimedium resources with consideration of the social status and persona of the agents. Furthermore, we had difficulty in arranging the agents on the screen in our implementations of some prototypes. These are important and practical issues and we are now formulating more precise research questions before initiating a sub-project in this area.

Multimedium Knowledge-Based Authoring Architecture

To efficiently develop and incorporate artificial intelligence techniques, a multimedium knowledge-based authoring architecture, OCTR-Tree, is currently being developed. Almost all sub-systems will use it to implement in a year. OCTR-Tree supports the OCTR model and adopts part of the Curriculum-Tree knowledge-based architecture features. Like Curriculum-Tree, OCTR-Tree is based on the assumption that the domain knowledge structure is isomorphic to the learning goal hierarchy which can be constructed through analysis. In this learning goal hierarchy or tree, each node can possess a cluster of learning episodes. An episode is a small unit of learning activity, usually having a beginning and an end. We can classify episodes into four types, namely, O-episodes, C-episodes, T-episodes, and R-episodes. These episodes are distributed over the tree and the whole discourse of activities is to sequence and run these episodes in the order: O-episodes, C-episodes, T-episodes, and finally R-episodes, in an iteratively spiral way. OCTR-Tree consists of five components: an OCTR-Tree editing environment, a prototype-based programming language, a class-based programming language, a tool set, and a multimedium object library.

Evaluation

Evaluation is an important part of the LISA project. A preliminary field evaluation has conducted on the learning effects of Distributed WEST and the result is positive and encouraging [Ch+ 92]. Most students think that it is effective to learn with a companion. To our surprise, we found that students seem to prefer competition to collaboration. This indicates that it needs more research to study the invaluable learning motives and striving attitude engendered by competition and try to avoid the ill effects, caused by revealing strengths and weaknesses in comparison.

For the languages we have designed, we have used them in three classes at National Central University and proved that they are achievable by the freshmen. Currently, we are comparing the differences between two classes; one uses these multiple languages and the other, like most traditional approach, uses a single imperative language, Pascal.

Also, we begin a comparison of social and self-directed learning and undertake a formal evaluation of Three's Company to test both motivational and cognitive effects in some controlled performance patterns; for example, the human student's performance is always between the two learning companions, like a sandwich.

Discussion

To develop systems for students to use for today, not for the next decade, we plan to set up an experimental classroom with over twenty Mac Quadra this Summer. We do not simply deliver our learning sub-systems to users and then evaluate [Clan 92, MuWo 92]. Students and teachers take part in the design process from the very beginning. Produce simple systems initially, then continue the development in the course of recurrent testing, observation, reflection, and redesign until it is able to be formally evaluated and used.

It is our hope that the project LISA would serve to stimulate further research interest and development effort in building effective multimedium learning environments. As the price of the computer is falling, the power of computer is increasing, and the technology of multimedium support is becoming more accessible, more than at any time we have an opportunity to create new breeds of learning environments that may bring important impact to education, even we believe that a revolution in education has to wait for a new business to market learning effective multimedium software that is profitable.

Acknowledgements

The author gratefully acknowledges the encouragement and support received from his colleagues and students in conducting this research.

Reference

[BuBr 76] Burton, R. R. & Brown, J. S. A Tutoring and Student Modeling Paradigm for Gaming Environments. In Colman, R., & Lorton, P. Jr. (Eds.), *Computer Science and Education*, ACM SIGCSE Bulletin, Vol. 8, no. 1, pp. 236-246, 1976

[Ch+ 92] Chan, T. W., Chung, Y. L., Ho, R. G., Hou, W.J. & Lin, G. L. "Distributed Learning Companion Systems -- WEST Revisited." The 2nd International Conference of Intelligent Tutoring Systems, *Lecture Notes in Computer Science*, 608, Springer-Verlag, 1992.

[Chan 89] Chan, T. W. Learning Companion Systems. PhD Thesis, Department of Computer Science, University of Illinois at Urbana & Champaign, 1989.

[Chan 91] Chan, T. W. Integration-Kid: A Learning Companion System. The 12th International Joint Conference on Artificial Intelligence, Sydney, Australia, Morgan Kaufmann Publishers, Inc., pp. 1094-1099, 1991.

[Chan 92] Chan, T. W. Curriculum Tree: A Knowledge-Based Architecture for Intelligent Tutoring Systems, The 2nd International Conference of Intelligent Tutoring Systems, *Lecture Notes in Computer Science*, 608 Springer-Verlag, pp. 140-147, 1992.

[ChBa 88] Chan, T. W. & Baskin, A. B. Studying with the Prince: The computer as a learning Companion, *Proceedings of International Conference of Intelligent Tutoring Systems*, 1988, June, Montreal, Canada, pp. 194-200, 1988.

[ChBa 90] Chan, T. W. & Baskin, A. B. Learning Companion Systems. In C. Frasser & G. Gauthier (Eds.) *Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education*, Chapter 1, New Jersey: Ablex Publishing Corporation, 1990.

[ChWa 92] Chan, T. W. & Wang W. C. Problem-Solving Model Approach for Teaching Multiple Introductory Programming Languages, *Proceedings of East-West Conference on Emerging Computer Technologies in Education*, Moscow, 1992.

[Clan 92] Clancey, W. J. Guidon-Manage Revisited: A socio-Technical Systems Approach. The 2nd International Conference of Intelligent Tutoring Systems, Lecture Notes in Computer Science, Springer-Verlag, pp. 21-36, 1992.

[CLLK 93] Chan, T. W., Lin, C. C., Lin, S. J. & Kuo H. C., OCTR: A Model of Learning Stages, World Conference on Artificial Intelligence in Education, August, Edinburgh, Scotland, 1993, to appear.

ERIC

[CoBN 89] Collins, A., Brown, J. S., & Newman, S. E. Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics. In L. B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser,* Hillsdale, NJ: Lawrence Erlbaum Associates Publishers, 1989.

[CuSe 90] Cumming, G. & Self, J. Collaborative Intelligent Educational Systems. In D. Bierman, J. Breuker, & J. Sandberg (Eds.), *Artificial Intelligence and Education*, pp. 73-80, Amsterdam: IOS, 1990.

[DiSe 92] Dillenbourg, P & Self, J. People Power: A Human-Computer Collaborative Learning System. The 2nd International Conference of Intelligent Tutoring Systems, *Lecture Notes in Computer Science*, 608, Springer-Verlag, pp. 651-660, 1992.

[GiSe 88] Gilmore, D. & Self, J. The Application of Machine Learning to Intelligent Tutoring Systems. In Self, J. (Ed.) *Artificial Intelligence and Human Learning, Intelligent computer-assisted instruction*, pp. 179-196, New York: Chapman and Hall, 1988.

[LiAS 77] Liskov, B., Atkinson, R., & Schaffert, C. Abstraction mechanisms in CLU. *Communications of the ACM* 20(8): pp. 564-576, 1977.

[Lieb 86] Lieberman, H. Using prototypical object to implement shared behavior in object-oriented system. *Proceedingsof the First ACM Conference on Object-Oriented Programming System, Languages, and Applications*, 1986.

[MuWo 92] Murray, T. & Woolf, B. Encoding Domain and Tutoring Knowledge via a Tutoring Construction Kit. *AAAI*, 1992.

[PaBr 84] Palincsar, A. S., & Brown, A. L. Peciprocal Teaching of Comprehension Fostering and Monitoring Activities. *Cognition and Instruction*, I(2), pp. 117-175, 1984.

[PaGM 91] Palthepu, S., Greer, J., & McCalla, G. Learning by Teaching. *The Proceedings of the International Conference on the Learning Sciences*, AACE, 1991.

[Piag 67] Piaget, J. *Biology and Knowledge*. Paris: Gallimard, 1967.

[RuNo 78] Rumelhart, D. E. & Norman, D. E. Accretion, Tuning, and Restructuring: Three Modes of Learning. In Cotton, J. W. & Klatzky, R. L. (eds.) *Semantic Factors in Cognition*, pp. 37-53, Hillsdale, NY: Lawrence Erlbaum Associates, 1978.

[Self 85] Self, J. A Perspective on Intelligent Computer-Assisted Learning. *Journal of Computer Assisted Learning*, Vol. 1, pp. 159-166, 1985.

[Self 86] Self, J. The Application of Machine Learning to Student Modelling. *Instructional Science*, Vol. 14, pp. 327-338, 1986

[Vygo 78] Vygotsky, L. S. *Mind in Society: The Development of Higher Psychological Processes* (Cole, M., John-Steiner, V., Scribner, S., & Souberman, E. Eds.). Cambridge, MA: Harvard University Press, 1978.

THE VISUAL INTERFACE:
Color & Location Coding In A Non-Linear Visual Display

WILLIAM W. CHAPMAN
Georgia State University
559 Willivee Drive, Decatur, GA 30033, USA

The traditional method of information dissemination is based on a linear form, and is reflected in such printed material as the history textbook or the daily newspaper. How we structure the information search is a very important part of our daily activities, and that includes the educational process. Often, the limitations of mass distribution guide our direction. Our search procedure follows the established linear form in print and the electronic medium. However, the application of multimedia in educational and consumer information presentations provides a much more complex form of distribution. Multimedia opens up new potential for the increasing quantity and complexity of information we must search and evaluate every day. To educate and inform effectively through the interactive media, we must first learn which presentation techniques help achieve this goal. Two of these techniques involve the use of location and color-coding in visual displays.

Interactive multimedia can be a powerful educator, however, in such instances when the only communication link to the student is through the visual display, the exchange occurs in a single mode. The visual interaction becomes the only means to receive information. Whether displays of user instructions, content, or learning example, all data must pass through the visual channel. This requires the observer to visually scan, process and identify every stimulus to determine its function and response. Therefore, nonverbal instruction requires the user to concentrate more on content. By following all the parameters of physical and psychological color perception, we can code a substantial amount of information into a visual learning experience. The medium can become the "teacher."

Consistency of location is a major part of our lifestyle, and the same concept applies to the visual presentation. Screen designs are extremely effective when information is displayed in a structured appearance, and expected information is where it should be . . . (Galitz, 1989). However, location can only be beneficial when relavent or supported by the task. Repetition and consistency of screen location have a positive effect on the time performance variable in a linear presentation. "A consistency of screen location for links has proven superior to the use of color in search tasks" (Christ, 1975; Aspillaga, 1991; Tsal & LaVie, 1988). Location was also superior to symbols, shapes and even color-coding in the processing of screen displays (Aspillaga, 1991; Tsal & LaVie, 1988). With a predictable location code, the search variable is basically eliminated. The student has learned the location of the target, and immediately examines the stimulus at that location. The process reduces to a simple identification task. As in most linear presentations, the ongoing task is a forward progression. Therefore, a consistency of location makes the entire search and identification process extremely efficient and effective.

In a non-linear presentation, a consistency of location targets may not always be possible, or practical. Different topics can create an entirely different set of targets and responses. The hierarchy and information flow can easily shift direction, or become totally new and unique.

While individually, each topic may follow a linear path, there is the ability to jump interactively between the visual design schema. Such non-linearity requires a new search task, which defeats the advantage of location coding. This study examines the effect of non-linearity in a visual sequence upon the search and identification process.

The other valuable technique for information display employs a color-coded scheme to establish relationships. Targets contain a particular color value that, in linear form, is consistent throughout a display series. Students are assisted in search and identification of stimuli by the color value, which can remain constant while the exact target may change by size, shape, text or any other manner. The color scheme can relate to a particular task, meaning or relationship. The biggest advantage to using color over achromatic is the opportunity to provide additional independent or redundant information to the student.

In many ways, color-coding is a much more versatile tool than location. The increased capability to create logical progressions, form group associations, define ideas by nature of the perceived or learned meaning of the color, or establish the link or pathway to a particular section, provides a clear advantage. Independent coding can identify links and define relationships. Redundant coding can perform those tasks as well as establish meaning, intervals and perceptions. The use of color as a steering device in printed or linear visual information can facilitate speed and accuracy of identification and enhance the short and long term cognitive effects (Christ, 1975; Green & Anderson, 1956; Lamberski & Dwyer, 1983; Smith, 1962). Students subjected to color-coded instruction and testing scored significantly higher than a monochromatic control group.

We must, however, use color wisely. Any learned meanings of color values will often supercede the desired response. Color values must be separated sufficiently to avoid confusion, unless such proximity establishes a scale or progression. The number and type of colors on a display will directly affect the performance of the presentation. With multidimensional displays, an increase in the number of non-targets containing the target color caused a substantial increase in the search time. Such an "effective density" or density of items displayed in the same color as the target, can cause havoc with the task (Cahill & Carter, Jr., 1976). In a display of multiple colors, search time increases with an increase in the density, or amount of different display colors (Smith, 1962). Therefore, when designing color displays, density is an important variable to effective communication. Optimal coding efficiency generally occurs when colors limit their range from four to seven.

ADVANTAGES OF COLOR AND LOCATION CODING

	Color	Location
Create Group Associations	•	•
Define Logical Progressions	•	
Decrease Search Time		•
Utilize Learned Meanings	•	
Establish Links or Pathways	•	•
Define Intervals or Scales	•	
Superior Processing Time		•
Create Perceptual Meaning	•	

Location and color-coding techniques enhance the effectiveness of linear presentations. In a non-linear format, however, consistency of location or color is not a guarantee. To handle a large amount of information, codes may require changing with the information. How will this effect an observer who randomly moves about a presentation? A group of codes at a particular location may change relationships or value as the display format alters. In color-coding, the change may be more severe. Coding that has established a meaning or direction may require a change with new displays. Often, the information flow dictates the code in use. If the flow changes, as in a non-linear format, the code changes. At this point, we would assume a decrease in the advantage of location or color-coding. There is also a strong possibility that non-linearity may eliminate the value of codes or even create a negative reaction.

As the use of multimedia grows in consumer, business and educational applications, the importance of effective display interaction will magnify. In display dynamics, the ultimate objective is one of performance. The speed and accuracy of the task, also short and long term cognitive effects become the goal of the multimedia designer.

To achieve this goal requires an interactive analysis of screen location and color-coding of stimuli in relation to response times in a non-linear visual display. This research will examine the ability of screen location, or target color when such location is not consistent, to cue the observer. A secondary question will examine location and color to determine if location remains a superior variable in a non-linear presentation. By examining the search times at non-linear points of a presentation, we can hypothesize certain results.

1) A consistent screen location stimulus improves the performance time of a non-linear interactive visual presentation.
2) Color-coded stimulus improves the performance time of a non-linear interactive visual presentation.
3) A consistent screen location stimulus is more effective than color-coding in improving performance time of a non-linear interactive visual presentation.

METHODOLOGY

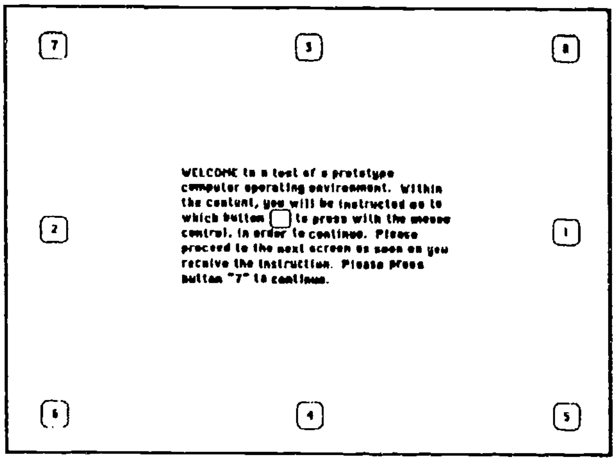
The presentation vehicle for this study consists of a series of visual displays presented in the HyperCard format on the Macintosh computer. The format resembles the typical on-line, instructional "help" documentation available to users.

Three variations of this tutorial were randomly administered to 82 participants. Research subjects included a group of university students as well as a group of multimedia professionals. All subjects are computer literate to the extent of understanding the mouse and target selection operation. Therefore, basic computer instruction was not necessary. Subjects averaged a current or completed college education.

To test the color and location variables, presentations were set-up in the achromatic control group, chromatic coding, and location coding variations. In a Macintosh operating system, applications or programs are stored in folders. The three versions, titled "Test 1", "Test 2" and "Test " were stored in an untitled folder. Therefore, the subjects had no indication of the particular test they were to perform. After launching the application, the only involvement with the participant was the test procedure.

The construction of the visual display involves two components. One element is the textual information that simulates the on-line instructional message. Information was transcribed from an Apple Computer, Inc. press release regarding the introduction of the new Powerbook Duo and Macintosh IIvx systems. The textual displays, edited into a form more suitable for visual displays, introduce observers to these new lines of Macintosh computers. This content was revised and condensed so that no individual would have previously worked with the material.

Each screen contains one paragraph with a single message content. The message structure and flow simulate a non-linear format. While the general topic remains the same, each screen is an individual idea with no linear relationship to another. The first screen of the display will carry an introductory message that includes instructions for determining the appropriate target when identified in the display text.



Screen #1 - Sample Layout of Text and Targets

The task in all instances involves moving the mouse cursor to the target and pressing the mouse button, a common Macintosh operation. Should the participant incorrectly respond to the directions of Screen 1, a secondary instruction reminder appears on the screen. Only after the observer understands the instructions and performs the correct response does the test continue. The last sentence of each message reveals to the user the identity of the target. This sends the user to the next screen display. An incorrect response returns a "beep" sound.

Nine screens comprise the presentation. Each display has a specific objective within the series.

Screen 1 - Introduction/Instruction
Screen 1A - Additional instruction
Screen 2 - Normalization (Initializes mouse/cursor position)
Screen 3 - Learn #1 - Level one of coding
Screen 4 - Data #1 - Level two of coding, measurement point #1
Screen 5 - Normalization - Disrupts linear code and reinitializes mouse
Screen 6 - Learn #2 - Level one of second coding
Screen 7 - Learn #3 - Level two of second coding
Screen 8 - Data #2 - Level three of second coding, measurement point #2
Screen 9 - Termination

At two points in the series, screens come up with a single message indicating the next target. These are the two data points where response measurements will be taken. The reason for including only a target identification on these two screens is simple. It avoids two problems with the methodology. One is the different reading and comprehension speeds of subjects. While the reading of a paragraph could vary significantly between respondents, the comprehension of five words is much shorter. Therefore, differences in skill should be minimal for the data screens. The relationship to the search task allows for a much more accurate evaluation of the variable. Two, it avoids any advantage should the subject begin reading the paragraph from the end, thereby revealing the target first.

The second important element of the display is the set of target and non-target items. Potential targets are numbered 1 - 8 for response identification, and placed in equally sized square boxes to establish the area for the cursor. Targets are located at 45° intervals around the periphery of the display. To maintain a non-linearity, numbers never repeat in the same location, and are never sequentially identified as the target. The displays may appear overly simplistic. This is necessary to limit the outside variables of human interaction.

The control group viewed the achromatic, randomly located target version. Targets were non-linear in location and identification, to avoid any pattern or repetition. To circumvent any grey-scale contrast influence, all values were either 100% black or 100% white.

The chromatic group viewed targets that were assigned one of eight color values. The identification of each target matched the achromatic version, allowing only for the color enhancement to affect the result.

To determine the target color values, a pre-test was performed on eight different hues. Colors were assigned by dividing a 360° color wheel into eight equal 45° sections. Beginning at 0°, or red, values were created at each 45° interval, keeping all other levels of luminance constant. The eight colors were then assigned to two different displays in an arrangement of targets similar to the layout of the original targets. The values were arranged so that adjacent colors on the wheel were not adjacent on the display to avoid any directional perception.

Sixteen respondents, viewing one of the displays, rated the eight values in order of dark to light. Two colors, orange and magenta, were perceived by the largest percentage as the middle values, and were assigned as the two target colors, with the other values representing the non-target colors. The first data screen of the main experiment references the orange value, while the second data point reflects a series of magenta coding. This change in the code also simulates a non-linear movement of the presentation.

The location coded version is similar to the monochromatic version except in the placement of the achromatic target. While the locations are consistent leading up to the two data screens, the target identification changes in the same sequence as the achromatic version. This isolates the location variable. To avoid any pattern, no target identifier repeats in successive screens, only the location. As in the chromatic version, the variable changes between points of measurement. In this instance, the location of the target shifts after the first data screen to simulate a non-linear direction.

RESULTS

The two independent variables, color and location, are hypothesized to effect a positive change on the search time of the non-linear variable. An application of the f-test should demonstrate any significant differences between the achromatic control group and the color or location variables. All data measurements, unless indicated, are in ticks, which is 1/60th of a second. Search times are based on the response time of the data measurement screens.

Target Search Times

	Variance	S.D.	Mean
Achromatic	4359.40	66.03	225.34/3.76 seconds
Location	11954.39	109.34	182.69/3.04 seconds
Chromatic	22743.41	150.81	291.73/4.86 seconds

Mean search times reveal an advantage to location coding (X=182.69/3.04 seconds) over color coding (X=291.73/4.86 seconds) or the achromatic version (X=225.34/3.76 seconds). As with linear presentations, location coding provides a faster search time for display targets. Color targets created the longest search times of the three non-linear versions.

An analysis of location coded stimuli revealed a significant difference to the achromatic group ($F_{1,110}=8.10$, $p<.01$). The advantage of location coding can reduce the search time variable of almost any achromatic presentation.

The difference in color coding is also significant, however, the effect is opposite of the expectation ($F_{1,108}=13.56$, $p<.001$). Search times took longer than the achromatic variable. This would indicate some strong limitations of color-coding in non-linear applications.

Obviously, an analysis of location and color codes revealed a significant advantage to a system of location stimuli ($F_{1,104}=27.63$, $p<.001$). Location remains a superior influence on search time in relation to color coding.

Reliability followed a slightly different sequence. Standard deviation for the achromatic variation (SD=66.03/1.10 seconds) was smaller than the location (SD=109.34/1.82 seconds) or color coded (SD=150.81/2.51 seconds) versions. Deviation levels increased with a change in the simulated non-linear structure.

A breakdown of data points measurements within each version reveals a different direction of deviation. The mean search time increased between the two points of the achromatic version. Both search time and reliability for achromatic targets became less effective during the presentation.

Comparison of Data Points Within Versions

	Variance	S.D.	Mean
Achromatic 1	2243.61	48.21	219.38/3.66 seconds
Achromatic 1	5810.67	77.58	234.76/3.91 seconds
Location 1	14106.64	121.03	211.37/3.52 seconds
Location 2	8278.37	92.72	154.00/2.57 seconds
Chromatic 1	25700.56	163.49	317.54/5.29 seconds
Chromatic 2	18607.89	139.11	265.92/4.43 seconds

Contrarily, the coded versions achieved more effective responses as the presentation progressed. Location improved response time and reliability. Color-coding, although not as effective, also improved student response. As reliability diminished for the achromatic targets, the coded variables increased. Search times also reflected better success during the presentations.

An analysis of measurements between data screens within each version indicates a significant advantage only in location coding. Even then, the confidence level suggests only minimal reliability ($F_{1,52}=4.94$ $p<.05$). Achromatic and color coded applications had no significant difference in the two areas of the presentation.

DISCUSSION

The advantage of location coding over color is evident in non-linear visual displays. The repetition factor is easily learned and applied from display to display. Once the method establishes, the concept carries along, eliminating the need to scan all targets. Should the previous location be incorrect, then a random search begins. Only then does the process return to the same level as other coded or uncoded displays. It can be suggested that, because of effective location coding, a relationship between content and location can be an extremely useful tool in non-linear presentations. There is some value noted in the cognitive effect of the location variable. The ability to learn location, in association with a redundant content selection, should produce a low search time compared to other methods of coding.

There is strong evidence to show that Hypothesis #1, proposing an advantageous value on location coded displays, is valid. A consistent location, even in a non-linear flow, provides faster search times. Within individual linear divisions, the location variable increases in value and importance by maintaining a content and response association.

The use of color coding provided an opposite, and unexpected result. Search times were significantly higher than either the achromatic or location version. Since the chromatic variable was the only addition to the achromatic version, this additional stimulus created a negative result. Once such cause for increased times could be the number of color values. Eight different hues exceed the recommended four to seven colors that most research indicates as optimum. The increased density produces several non-targets that must be processed by the observer. It could be suggested that all non-targets must be evaluated against the target to determine the accuracy of the target before response. If valid, this argument also could support the lack of any learning effect between the two data points of the chromatic version. However, there may be other factors that influence the outcome.

As a repetition of location created a "probable" target that assisted the search task, location also may influence the search point of the next color target. Observers, when presented with a new display, may cognitively return to the original location as a starting point for the search, thus giving the advantage to location coding, which the results indicate. In the same schema, the student may first search the "probable" location, and determine its color value. When the target color does not appear at the "probable" location, or is not the necessary response, the search begins again, as in all variables. Therefore, it becomes a game of chance, in which the odds of location, always being the first choice, come out ahead. The significant negative outcome of Hypothesis #2 demonstrated the reverse effect of color-coding in relation to a non-coded display.

The method of coding is not always apparent to the user. In all three variables, prior knowledge of the system was not indicated, or suggested, by the structure. Therefore, some influence may have occurred in the realization of a coded sequence. One would suggest the advantage belong to any coded system.

The location coded displays were significantly faster than the color-coded versions ($F_{1,104}=27.63$, $p<.001$). Under the conditions of this study, we can safely assume the increased performance value of location coding. Hypothesis #3 presents a useful guideline when designing this type of visual presentation.

Each test can be separated into two measurement groups. The first point follows two stimuli, the second after a change of target and three displays. Only in the location study did any significant learning curve appear ($F_{1,52}=4.94$, $p<.05$). Location responses significantly improved as the coding scheme progressed. Color results, although demonstrating a similar trend, were not large enough to support such an argument. The increased density may require more repetition to be effective. The achromatic revealed negative results. While no conclusion can be reached, it does open an interesting avenue for further research into coded patterns with repetitive non-linear displays.

From the results we can suggest certain guidelines regarding coding in a non-linear format. After an evaluation of the task to be performed, a criteria of performance objectives can assist in a coding objective. Results indicate search times most suitable to a location coded system. Such a system may be valuable when students need to retrieve or search information quickly. The advantage of speed enhances volume of information and economics of operation.

Color coding does not perform well under these circumstances. Future research must explore the ability of color in non-linear application to perform other functions. Redundant coding in conjunction with location may provide additional advantages, as search times could be supplemented with additional data based on color. Educational presentations are the likely candidates for supplementing information with color-coded stimuli. The tools of learning can easily be incorporated into redundant and repetitive schemes. The ability to associate color with content or learned meanings also can work with location and/or content directions. While no conclusion on learning effects could be reached, this study illustrates the potential for cognitive processes to have an influence on repetitive non-linear presentations. Future research also can explore these combinations over a series of non-linear displays with varying levels of repetition.

One reason for longer color coded search times involves the display density. Eight values were displayed, more than the recommended value in linear forms. By applying the four to seven values limitation of linear approaches, we can evaluate different density variables. It could be speculated that any introduction of color has a negative effect, however, I speculate that color will show similar characteristics to linear forms in non-linear formats. Research also can examine the ability to multi-code color and content. The effects of redundant color coding may prove superior to location when the evaluation of content becomes a more important task.

The characteristics of color and location coding suggest similar advantages and disadvantages between non-linear and linear presentation. By further research into the limitations of coding variables, we can design and implement better educational and information interactive presentations based on their strengths and combined advantages.

REFERENCES

Aspillaga, M. (1991). Screen Design: Location of Information and Its Effects on Learning. *Journal of Computer-Based Instruction,* 18(3), 89-92.

Cahill, M. C. & Carter, Jr., R. C. (1976). Color Code Size for Searching Displays of Different Density. *Human Factors,* 18(3), 273-280.

Christ, R. E. (1975). Review and Analysis of Color Coding Research for Visual Displays. *Human Factors,* 17(6), 542-570.

Galitz, Wilbert O. (1989). *Handbook of Screen Format Design.* Wellesley, MA: QED Information Sciences, Inc.

Green, B. F. & Anderson, L. K. (1956). Color Coding in a Visual Search Task. *Journal of Experimental Psychology,* 51(1), 19-24.

Lamberski, R. J. & Dwyer, F. M. (1983). The Instructional Effect of Coding (Color and Black and White) on Information Acquisition and Retrieval. *ECTJ,* 31(1), 9-21.

Smith, S. L. (1962). Color Coding and Visual Search. *Journal of Experimental Psychology,* 64(5), 434-440.

Tsal, Y. & LaVie, N. (1988). Attending to Color and Shape: The Special Role of Location in Selective Visual Processing. *Perception and Psychophysics,* 44(1), 15-21.

ERIC

A Framework for Delivering Large-Scale Hypermedia Learning Material.

HUGH DAVIS, GERARD HUTCHINGS AND WENDY HALL.
Department of Electronics and Computer Science
The University of Southampton, Southampton SO9 5NH, UK.
e-mail hcd@ecs.soton.ac.uk, tel +44 703 593669, fax +44 703 592865

Introduction

This paper discusses a resource-based approach to presenting learning materials. A platform for delivering such materials is introduced, and our early experiences of using this system in a teaching environment are summarised.

In the past, the time required to create computer assisted learning (CAL) material has limited its introduction into mainstream education. Typically 100-150 hours has been cited as the time required to produce one hour of instruction (Christie, 1990). Also, material produced in this way is difficult to update and becomes redundant as knowledge changes. Furthermore, no two teachers have exactly the same views on how a subject should be taught, which means that CAL material is not often used outside of the institution in which it was developed. A resource-based approach integrating current hypermedia practices can address all of these problems by permitting the creation and integration of new material and the customisation and easy updating of existing material. In this way it is possible to overcome the limitations of traditional CAL which have prevented the widespread use of computer-based materials in education.

In 1992 the U.K. Universities Funding Council (UFC), announced that it was making provisions for the development and delivery of computer based learning materials for higher education under the Teaching and Learning Technology Programme (TLTP). The aim of the programme was "to make teaching and learning more productive and efficient by harnessing modern technology" (UFC, 1992) and builds on previous U.K. funding initiatives such as Computers In Teaching Initiative (CTI) and Information Technology Training Initiative (ITTI). Funding was provided for subject based consortia, which consist of departments from several institutions working together to create courseware for a particular subject, and institutional bids, which involve the creation and delivery of courseware as part of an institutional infrastructure with backing from senior management. A number of the successful bids involved the use of Microcosm (Davis, Hall, Heath, Hill & Wilkins, 1992; Fountain, Hall, Heath & Davis, 1990), an open hypermedia system, as the delivery platform: in particular the University of Southampton was awarded a sizeable grant to integrate the use of technology into the teaching on a number of high profile undergraduate courses, using Microcosm as the central platform.

This paper describes the Microcosm model and philosophy, and explains the advantages of taking an open, resource-based approach to the development and delivery of learning materials over traditional CAL practices. We describe early experiences in creating materials for the Microcosm environment, paying special attention to the features of the system which make authoring in this way different from traditional practices. Finally, we look at the initial use of Microcosm by students of History and Biology at the University of Southampton.

The Microcosm Model and Philosophy

Microcosm consists of an open message passing system combined with a number of viewers and filters. Viewers are programs that can display the various formats of text documents, pictures, sound and video. They can communicate with the filters to establish what active areas exist in a particular document, and allow users to make selections within the document, then choose actions such as following a hypertext link. The information concerning the selection and action are then bundled into a message which is sent through a chain of filters, each of which has the opportunity to respond to the message by taking some action. Particularly important filters are the *linkbases* which contain all the information about links and which can respond by offering the user the opportunity to display the document which is the destination of the link. Other filters provide aids to navigation, dynamic link services and link creation facilities.

In brief the advantages that such a system offers when producing learning materials are as follows:

a. No Mark-up in Documents

All the information concerning links is held in link databases or linkbases. This means that by installing different linkbases, as appropriate, it is possible to provide different views on the same set of documents. It also means that it is possible to continue to view the document using the application that created it, which is necessary to keep the system open and to allow inter-application hypermedia functionality.

b. Open Architecture and Connectivity to Other Applications

It is possible to make links that dispatch other programs with a given dataset as a Microcosm viewer. More importantly it is possible to follow links from third party applications. In the Microsoft Windows version of Microcosm this may be achieved via the DDE where the third party application supports such communications, or else via the clipboard. The process of adapting an application to talk to the Microcosm message system typically involves writing a few lines of code in the application's macro language: currently we have added such functionality to Word for Windows, Toolbook, the SPANS Geodata system, Superbase, MS-Access and Autocad, as well as producing our own specialist viewers for around ten other common formats such as text, bitmaps and Windows Meta-files.

If learning at a computer is to be about more than simply absorbing information and responding to set questions, then the system must offer both the author and user a seamless interface to the normal applications and tools that make up the entire system; for example a mathematician may well wish to use a package to investigate the shape and behaviour of a function. In such cases it should be possible to link into and out of the required package.

Since the architecture and message system of Microcosm are open it is very easy to extend the functionality of the system to suit the needs of a particular application area: furthermore it is possible to distribute Microcosm functionality across different machines on a network, even where the machines themselves have differing architectures. Unix and Macintosh versions of Microcosm are currently under development.

c. Reduced Authoring Effort.

In most hypermedia systems links have specific source and destination anchors. Microcosm supports such "specific" links, but also supports more general links. "Generic" links have a fixed end point, but may be followed from any point where a given selection occurs within a document or specified set of documents. This means that it is possible to create a single link to a piece of reference material, which may then be followed wherever the specified source selection occurs without having to re-make the link at every possible source point. This facility makes it possible to put together sets of reference material on a particular subject domain, and allow users access to this material from whatever point in whatever document they are currently examining. For example, we have made good use of a Dictionary of Biology, by automatically generating a set of generic links to each term in the dictionary. Now, by intalling this linkbase along with the others in any application in the biology domain, the user has immediate access to all this information from any document.

There are a number of other ways of following links in Microcosm. Documents may have associated keywords which may then be used for link following. Also text documents may be pre-indexed to allow a built-in information retrieval system to attempt to match suitable documents or sections of a document from a query made from a text selection (Li, Davis & Hall, 1992). Authors find this facility particularly useful when attempting to produce links in very large bodies of material, and this feature is a fundamental tool for deducing relationships between various documents where no appropriate links have been previously manually authored.

d. Selective Browsing.

Both links and documents in Microcosm may have user defined attributes. These attributes enable us to attach keywords and descriptions to the documents and links. These attributes allow users to make queries of the system. It is possible to navigate the document collection entirely by document attributes.

The resulting Hypermedia system has an intrinsically different feel from most other Hypermedia systems. In effect Microcosm provides a link service which supplements the normal navigational facilities provided by the operating system, resulting in an environment in which the onus is on the students to interrogate the system in order to find answers to questions which are formulated either by themselves or by a tutor - i.e. the student must ask the system "what other information do you have about subject X ?"; most other systems take the opposite approach - i.e. they announce to the student "I have information on subject X", thereby reducing the degree to which the student dictates the course of the interaction, and reducing the requirement for the student to formulate and ask questions. This free-ranging environment is ideal for providing fast access to large volumes of multimedia material, as is often required in conducting research, but may however be either intimidating for naive users, or inappropriate for some teaching needs. In these cases, more formally defined routes through a body of information are required, which are provided in the form of Mimics. These are tours through a particular set of multimedia documents which may be defined by an author, and which may be followed by the reader, but without sacrificing the facility of being able to follow other hypertext links as and when required.

The requirements of the various TLTP projects

Microcosm is quite unlike packages such as Toolbook and Authorware. These packages concentrate on providing very smart delivery of carefully prepared material: Microcosm concentrates on allowing users to browse through large bodies of material in a seamless manner. Because of this fundamental difference we have only a little experience of the sort of techniques that authors will use in delivering teaching materials using the system. As the TLTP programme is getting under way and a number of lecturers in a number of subject areas are starting to produce delivery materials using Microcosm, we are beginning to get feedback on the sort of special facilities that authors require.

Principally we have found that our user community has broken down into two distinct groups. The first group wishes to use Microcosm as the entire delivery mechanism. The second group wishes to use some other package (Toolbook, Authorware and Guide are notable examples) for the front end delivery of the teaching materials, but wishes to use Microcosm for linking together the materials. The advantages of this approach are:

a. Microcosm may be used to launch any application. A common requirement is to launch one application to simulate an experiment, while simultaneously launching a hypertext with explanatory notes.

b. Once Microcosm is running it can provide a link service over and above that provided by closed hypermedia systems. It is possible to follow links from one package through to another package. A common example of this is to have a set of reference material defining terms that will be used throughout a body of teaching material. Using Microcosm's generic links it is possible to follow links to the reference material from whichever package the student is currently working.

c. Microcosm allows separate logins for each user. (Windows and DOS do not have any concept of different users.) This means that Microcosm can keep track of who is using the system, and allow staff to configure the system in different ways for different users.

d. All Microcosm messages may be monitored. It is therefore possible to produce statistics of what information is viewed, and how that information is accessed.

In section 2 it was pointed out that the open architecture makes it easy to add new functionality. There have been three notable requests for enhancement of the basic model as described in the previous section.

a. A student progress logging system

Much importance has understandably been attached to measuring student progress through a body of materials. Microcosm has a logging filter, which saves a copy of every message that is sent through the filter chain. This allows open ended analysis of user interactions with the system. Teachers have asked us to provide specific tools to analyse these log files.

b. An event monitoring system with active buttons.

Currently Microcosm actions and buttons are actuated explicitly by the user. However, authors have identified occasions when they wish to make Microcosm take some action when a specific event occurs. An example of this is the request to load a text file to the screen at some specific point during the play of some piece of music or some video. These examples are being dealt with by introducing the idea of *automatic links*, similar to those described in Palaniappan, Yankelovich & Sawtelle (1990), into the appropriate viewers. More problematic have been some requests to monitor events in applications which are not Microcosm aware and cannot be programmed to send messages when the required event occurs.

c. *Extended Mimics.*

The current Mimic system allows an author to define a linear route through a set of documents. The author has limited control over the appearance of the document. There is a demand for greater control of the desktop from the Mimics, so that the author can control exactly what appears in front of the user. This is particularly important when a student is being introduced to the system, and we are attempting to provide such control initially by a simple scripting language. Later we will provide a graphical interface for the author to produce such scripts. A further enhancement we hope to introduce to the Mimic system involves allowing various branching and looping structures.

Experiences of Authoring

Many of the problems which are experienced when using hypermedia are in part due to the fact that not much serious authoring has been done on material which is intended for users rather than on material which is intended for research purposes (Hutchings, Hall, Briggs, Hammond, Kibby, McKnight & Riley, 1992). However, some work has been done on hypertext authoring and patterns are beginning to emerge about how authors go about creating hypermedia learning materials (Kidd, Hutchings, Hall & Cesnik, 1992). Previous experience indicates that the authoring process is as follows:

a. define the content in general terms
b. create a pictorial overview of the main topics
c. begin to create material
d. level of detail increases
e. new links and content are incorporated into structure
f. refine and update as you go

It is important to note here that whereas in the past content has been defined completely before any programming or work on the computer began, with the highly interconnected information environment that is possible with hypermedia systems, the design process is an ongoing one, with the need for new links and content areas only becoming apparent during the process of authoring.

Establishing a resource-based approach to the creation of learning materials is different again from the task of authoring closed hypermedia applications. It comprises two parallel and continuous processes. First, the resource material must be collected together and made available in a suitable format. Links are created between individual pieces of information to create what might be referred to as 'raw hypermedia'. However, as has been pointed out before, this is not a suitable environment for learning to occur (Hutchings, Hall, Briggs, Hammond, Kibby, McKnight & Riley, 1992) . Educational hypermedia must support a variety of purposes, from providing introductory material for naive learners, through to a general information resource for experienced learners or even those who are experts in their particular field.

The second stage is therefore one of refining or tailoring the resource-base to meet specific needs, for example by providing guided tours, or creating alternative link sets for different groups of users - one set for introductory learning, another more extensive set for experienced learners. Thus the same basic material is available to both groups, and inexperienced learners have access to the same information as experienced learners, but the complexity of the overall resource-base is hidden from them until they can 'find their feet' in the system.

First Experiences of Using Microcosm in a Learning Environment

At the time of writing this paper we have completed three surveys with students, the first of which involved using Microcosm to store and deliver notes for a course introducing computer programming (Davis, Hall, Hutchings, Rush & Wilkins, 1992). There were about 50 students on the course and they were required to use the system for at least one session in order to complete some coursework: many used the system more than this. From this exercise we learned important lessons about how to present the features of Microcosm to the user, and also about how to log the system use so that all a complete reconstruction of a session would be possible. These lessons have been fed back into the system, and we are now piloting work for the TLTP project. Currently the system is subject to much more extensive use by both students and courseware developers.

Another group of users of the system have been History students (Colson & Hall, 1991), and this year they have produced dissertations in Microcosm. *Cell Motility*, an application for undergraduate Biology students developed using StackMaker (Hutchings, Carr & Hall, 1992) a Macintosh based toolkit, has been extensively tested (Hall, Thorogood, Hutchings, & Carr, 1989; Hall, Thorogood, Sprunt, Carr & Hutchings, 1990; Hall, Hutchings, Carr, Thorogood & Sprunt 1993; Hutchings, Hall, Colbourn, 1993). The content and structure of this application has been ported to Microcosm, and this system is being used with undergraduate biology students for first time this year, and students' reactions to the system have been interesting (Hutchings, Wilkins, Weal & Hall, 1993). Responses regarding the effectiveness of the system were encouraging: 80% of users enjoyed using Microcosm, 88% felt that Microcosm was an effective learning resource, and 85% said they would use it again as a general source of reference. However, only 65% disagreed with the statement "This 'high-tech' route to information is intimidating", suggesting some unease with the Microcosm environment. 82% found manipulating windows easy, so the intimidation must have been due to factors other than the multiple windowing environment. Only 9% of students using the original StackMaker version of Cell Motility said they felt intimidated by the technology, and since the two systems offer essentially equivalent functionality, the cause of this problem is not immediately apparent. However, the overall reaction to Microcosm was extremely positive.

As the TLTP projects continue to prepare courseware we will gain further experiences of how authors wish to use Microcosm, and how we can help them to make best use of the system. A large part of the problem that we face in introducing such a system into the university is in overcoming entrenched attitudes. Many believe that prior failures using traditional CAL demonstrate that technology offers no added value in teaching, at least at university level, and others have fixed ideas about the sort of delivery system they wish to use and are unwilling to explore other routes.

Conclusions

In the future there is a good chance that a large proportion of learning in higher education will be via technology based systems: the fact that the UFC has initiated the TLTP in the U.K. is evidence of this. If this endeavour is to succeed, the obstacles encountered with traditional CAL - large authoring times, difficulties with updating and personalising material, the prescriptive nature of the delivery medium, etc. - must be overcome. We believe that a resource based approach such as that encouraged by Microcosm will provide a basis from which many of these problems will be solved.

References

Christie, A. (1990). The use of Interactive Videodisc in the Teaching of Orthopaedics in Physiotherapy, *Medical teacher* 12(2), 175-179.

Colson, F. & Hall,W. (1991). Multimedia Teaching with Microcosm-HiDES: Viceroy Mountbatten and the Partition of India, *History and Computing* 3(2), 89-98.

Davis, H.C., Hall, W., Heath, I., Hill, G. & Wilkins, R. (1992). Towards an Integrated Information Environment with Open Hypermedia Systems. *In:* D. Lucarella, J. Nanard, M. Nanard, P. Paolini. *eds. The Proceedings of the ACM Conference on Hypertext, ECHT '92 Milano,* ACM. 181-190.

Davis, H.C., Hall, W., Hutchings, G., Rush, D. & Wilkins R. (1992). Hypermedia and the Teaching of Computer Science: Evaluating an Open System. *In:* David Bateman and Tim Hopkins. *eds. Developments in the Teaching of Computer Science*, The University of Kent.

Fountain, A.M., Hall, W., Heath, I. & Davis, H.C. (1990). MICROCOSM: An Open Model for Hypermedia With Dynamic Linking, *in* A. Rizk, N. Streitz and J. Andre *eds. Hypertext: Concepts, Systems and Applications. The Proceedings of The European Conference on Hypertext, INRIA, France.* Cambridge University Press.

Hall, W., Thorogood, P., Hutchings, G. & Carr, L. (1989). Using Hypercard and Interactive Video in Education: An Application in Cell Biology. *Educational and Training Technology International* 26(3), 207-214.

Hall, W., Thorogood, P., Sprunt, B., Carr, L. & Hutchings, G. (1990). Is Hypermedia an Effective Tool for Education? *In* McDougall, A. & Dowling, A. *eds. Computers in Education,* Elsevier Science Publishers B.V.: North-Holland pp. 1067-1074.

Hall, W., Hutchings, G., Carr, L., Thorogood, P. & Sprunt, B. (1993). Interactive Learning and Biology: A Hypermedia Approach *In* Ferguson, D.L. *ed. Advanced Technologies in the Teaching of Mathematics and Science*, Springer-Verlag: Heidleberg. (In Press)

Hutchings, G., Hall, W., Briggs, J., Hammond, N.V., Kibby, M.R, McKnight, C. & Riley, D. (1992) Authoring and Evaluation of Hypermedia for Education, *Computers in Education* 18,171-177.

Hutchings, G., Carr, L. & Hall, W. (1992) StackMaker: An Environment for Creating Hypermedia Learning Material. *Hypermedia* 4(3), 197-211

Hutchings, G., Hall, W & Colbourn, C.J. (1993). Patterns of Students' Interactions with a Hypermedia System. *Interacting With Computers*. (in press)

ERIC

Hutchings, G.A., Wilkins, R.J., Weal, M. & Hall, W. (1993) Microcosm: Real World Use. *Department of Electronics & Computer Science, University of Southampton, U.K., CSTR 93.*

Kidd, M.R., Hutchings, G., Hall, W & Cesnik, B. (1992). Applying Hypermedia to Medical Education: an Author's Perspective. *Educational and Training Technology International* 29(2),143-151.

Li, Z., Davis, H.C. & Hall, W. (1992). Hypermedia Links and Information Retrieval. *The Proceedings of the 14th British Computer Society Research Colloquium on Information Retrieval*, Lancaster University.

Palaniappan, M., Yankelovich, N. & Sawtelle, M. (1990). Linking Active Anchors: A stage in the Evolution of Hypermedia. *Hypermedia* 2(1).

UFC (1992) *Universities Funding Council, Teaching and Learning Technology Programme Universities Funding Council Circular Letter 8/92*, available from Universities Funding Council, Northavon House, Coldharbour Lane, Bristol, BS16 1QD, 1992

Evolving from Multimedia to Virtual Reality

CHRISTOPHER J. DEDE
Center for Interactive Educational Technology
George Mason University, Fairfax, VA 22030, USA

Multimedia must bridge from its current role of augmenting data delivery in conventional instruction to instead fostering a new model of teaching/learning based on learners' navigation and creation of knowledge webs. Such a transformation requires evolving today's often fragmentary multimedia applications into more structured inquiry approaches that build on web-like architectures from hypermedia. As a further stage of development, through advances in visualization and virtual communities, multimedia can become the basis for rich virtual "worlds" that provide both intellectual and emotional stimulation.

Two stages of multimedia's potential evolution involve incorporating hypermedia to enable knowledge construction by learners and using visualization and virtual communities to create artificial worlds. Such advances would make multimedia the core of an information infrastructure that could be a driveshaft for educational reform. Without such a transformation, multimedia risks continuing its present status as a hood ornament for the conventional classroom, contributing some motivation at the risk of causing intellectual indigestion through information overload.

Multimedia and Hypermedia

Using decontextualized learning-by-telling pedagogical strategies to prepare students for a world that is increasingly complex and culturally diverse is dysfunctional. Media-based, situated learning-by-doing is an emerging instructional paradigm essential for leveraging major increases in educational effectiveness. Much recent work in cognitive science has centered on "cognitive apprenticeships," characterized by student activity in mastering authentic activities rather than in solving decontextualized problems such as those at the end of textbook chapters (Brown, 1989). Resnick (1987) describes ways in which one can design learning environments so that they build on people's abilities in real world settings, enabling them to more easily master new competencies without formal instruction.

Multimedia technologies have great potential to empower learners' mastery of authentic activities. The leverage that sophisticated multimedia provides stems from a synthesis of multiple attributes rather than any single characteristic: learning via structured discovery; motivational power; ability to tap multiple learning styles; web-like representations of knowledge; enhanced mastery through learner authoring of materials; the collection of rich evaluative information; technology-supported collaborative inquiry. At a time when the educational reform movement is providing momentum for change, the sophisticated user-tailorable multimedia devices that are necessary to support these instructional attributes are finally becoming affordable by schools. The availability of this technological infrastructure enables extending beyond current multimedia implementations to new types of instructional strategies that take advantage of multimedia's unique capabilities.

Elsewhere in this *Proceedings*, our research group's work at George Mason University on designing a multimedia prototype to foster higher-order thinking skills is described (see "Multimedia, Constructivism, and Higher Order Thinking Skills"). Our ultimate goal is to build a generic instructional shell for thinking skills that can be easily customized to multimedia content across a range of disciplines, from science and mathematics to the social sciences and humanities. As an initial step toward that objective, The Civil War Interactive Project uses Ken Burns' documentary series on PBS, *The Civil War*, as the core of its multimedia database. With short-term funding from the Corporation for Public Broadcasting and George Mason University and with limited technical assistance from Apple Computer, our team has produced a design demonstration for a computer-based instructional system that teaches a structured process for higher-order thinking while learners engage in guided historical inquiry.

Hypermedia—the associative, nonlinear interconnection of multimedia materials—extends this type of sophisticated multimedia in several ways (Dede, & Palumbo, 1991):

- The associative, nonlinear nature of hypermedia mirrors the structure of human long-term memory, lessening users' need to map from how computers represent data to how people store information.
- The capability of hypermedia to reveal and conceal the complexity of its content lessens the cognitive load on users of this medium, thereby enhancing their ability to assimilate and manipulate ideas.
- The structure of hypermedia facilitates capturing and communicating knowledge, as opposed to fragmented data, allowing users to view their own mental models as visual webs of nodes/links.
- Hypermedia's architecture enables distributed, coordinated interaction, a vital component of teamwork, organizational memory, and other "group mind" phenomena.

Multimedia systems that store the pattern of users' traversal through the database, such as our projected thinking skills shell does, enable learners, teachers, and researcher to track students' progression through knowledge webs in the content material. This allows more reflective learning, as well as sophisticated evaluation strategies that trace shifts in students' patterns of thought.

The ultimate potential of hypermedia is to evolve beyond knowledge representation to knowledge construction, so that the learner can modify/add nodes and links. Few hypermedia systems provide tools to support this type of activity, yet these capabilities are crucial to advance students past passively assimilating knowledge to actively creating their own mental models. Research and development in this area is essential to shifting multimedia from a visual database to a knowledge construction set. Such an evolution is crucial to the development of evaluation systems that move beyond measuring factual retention into assessing mastery of higher order cognitive skills.

In Schrage (1990), Alan Kay analyzes the implications of new media through the question, "What does a medium ask you to become in order to use it?" Print requires a rational reader; television, a passive observer; the telephone, a conversationalist. When structured into learning environments that motivate guided inquiry, hypermedia has the potential to develop more user metacognition (thinking about thinking) than linear media. While reading, listening, and viewing are passive in linear media, web structures demands continuous choice and navigation on the part of the learner. This strength of hypermedia representations provides a bridge for transforming fragmentary multimedia databases into richly detailed virtual worlds.

Virtual Realities

Trainers have long used technology-based simulations to mimic simplified real-world environments. However, virtual realities are qualitatively different than small-screen models because entering and exploring a "world" is a more intense mode of human-machine communication than "conversational" interfaces in which the user commands the computer to do something, then waits for a reply (Walker, 1990).

The key capability that artificial realities for learning add to current simulations and microworlds is immersion: the subjective sensory impression that the user is "inside" a synthetic environment comprehensive and realistic enough to induce the willing suspension of disbelief. Much research is needed to clarify all the variables that affect this sense of immersion. At present, most work by investigators studying artificial realities centers on achieving sensory (predominantly visual) fidelity to shift the user's frame of reference and sense of personal space. However, this focus on objective realism neglects other, more subjective dimensions that underlie immersion. For example, dreams are immersive even though their sensory and epistemological fidelity may be low. As another illustration, a well-written horror novel read alone at night can induce strong suspension of disbelief even though no visual images are utilized.

A weak analog to immersion interfaces that many readers will have experienced is the IMAX theater, in which a two-story by three-story screen and high resolution images can generate in the observer strong sensations of motion. Adding three-dimensionality, highly directional and realistic sound, an even wider visual field, and the ability to interact with the virtual world through natural physical actions produces a profound sensation of "being there," as opposed to watching. Because common sense responses work in artificial realities, the learner quickly develops feelings of mastery, rather than the perception of helplessness and frustration typical when first attempting to use a new information tool.

Also, these types of educational environments are strongly motivating, since designers of virtual worlds can readily incorporate three powerful levers for emotional involvement: fantasy, challenge, and curiosity (Malone, & Lepper, 1985). Artificial realities can be tailored to the affective needs of individual learners, incorporating fantasy characters matched to the student's personality in challenging games or complex adventure-worlds. Of course, these fantasy settings must be carefully designed to suppress the learning of behaviors harmless in a virtual environment, but hazardous in the real world (e.g. walking through a buzz-saw).

In addition to providing through immersion greater mastery and motivation than simulations, artificial realities can support two types of "magic" strongly conducive to learning: visualization and virtual communities. Virtual environments that simply mimic reality have their uses, such as allowing learners to experience activities that are dangerous or expensive in the real world. However, adding the ability to magically act in ways impossible in the real world opens up new dimensions for education. Through visualization, learners can manipulate typically intangible entities such as molecules and mental models; through virtual communities, students can interact in rich psychosocial environments populated by simulated beings.

Visualization

Imagine a medical trainee in an artificial reality entering a room labeled "Laboratory." Inside are three types of tangible, manipulable objects. First, the learner can explore the uses of commonplace laboratory devices such as microscopes and centrifuges. Second, the learner can manipulate typically intangible physical objects such as molecules, altering size to perceive in detail their three dimensional configurations and experimenting with maneuvering two molecules together to understand how one catalyzes a change in the other. Third, the learner can perform similar actions with typically intangible cognitive objects, such as mental models or knowledge structures, looking for geometric patterns that expose the similarities and differences of contrasting theories.

The term applied to rich artificial realities for visualizing information is "cyberspace," from a science fiction novel by William Gibson (1984). The vision of our civilization a couple decades hence that Gibson presents is both intriguing and plausible. The "nervous system" of global business is based on workers manipulating huge virtual structures of data in a shared artificial reality; teleoperation (performing activities over distance) and telepresence (mimicking face-to-face contact through video/graphics representations) dominate human activity. Some "cybernauts" are interested in having life imitate art and are building computational tools that would enable Gibson's imagined future.

The fundamental idea underlying visualization is that of displacing cognitive complexity into the human visual system. Human beings have very powerful pattern recognition capabilities for images. Spatial data management systems and scientific visualization have established that—when symbolic meanings are mapped into visible attributes such as shape, texture, size, color, and motion—increased insight into underlying structural patterns of information is attained (Tufte, 1990).

The virtual medical laboratory outlined above would support two types of visualization embedded in an artificial reality: "sensory transducers" that allow users' eyes, ears, and hands to access previously imperceptible phenomena (such as a molecule) and "cognitive transducers" that perform a similar function for intellectual entities. Sensory transducers provide a means of grasping reality through illusion (Brooks, 1988). Using computers to expand human perceptions (e.g. allowing a medical student—like Superman—to see the human body through X-ray vision) is a powerful method for deepening learners' intuitions about physical phenomena.

Realistic, directional sound is attainable today to enable auditory transducers, and visual transducers already exist through computer graphics. However, tactile forcefeedback is a difficult and expensive hardware challenge; for the early 1990s, cost will likely restrict haptic transducers to applications, such as exploring molecular docking sites, that necessitate people's very delicate and important sense of touch (Minsky et al, 1990). Together, all these types of sensory visualization modalities can combine to intensify immersion in a magical environment of previously intangible physical entities.

A second form of visualization, cognitive transducers, make intellectual entities such as knowledge structures visible and manipulable. They are a logical extension of sensory transducers in enhancing the power of artificial realities for education. Transforming the symbolic into the geometric via data visualization is useful in situations where the amount of data is large and interacting with the data to shape its presentation can aid in interpretation.

A leading-edge illustration of generic data visualization approaches is the Information Visualizer, an experimental interface that uses color and three-dimensional, interactive animation to create information objects (Robertson, Card, & Mackinlay, 1991). Designers of educational "microworlds" (simulations in which the user can change the rules by which the virtual environment functions) also frequently incorporate cognitive transducers. For example, the Alternate Reality Kit allows the user to see and manipulate abstractions such as Newton's Law of Gravity (Smith, 1987).

Beyond visualization, a second type of artificial reality magic is virtual communities. Learners can interact in rich machine-mediated psychosocial environments populated both by videolinks to other people and by simulated beings. These simulated beings may be avatars (computer graphics representations of people) or knowbots (machine-based agents); each adds an important dimension to education in artificial realities.

Virtual Communities, Knowbots, and Avatars

Enabling people to communicate effectively across barriers of distance and time by forming virtual communities is an emerging goal for information technology. During the last decade, the field of computer-supported cooperative work (CSCW) has established that developing productive technology-mediated human interaction involves complex psychosocial issues that extend well beyond earlier models of simply transmitting data along a channel from sender to receiver (Greif, 1988). Communication depends on affective as well as cognitive interchange; researchers studying collaborative educational environments are just beginning to conceptualize the complicated emotional dynamics of peer teaching and learning.

During the 1990s, wide-area, broadband information infrastructures such as the National Research and Education Network (NREN), will increasingly link educational sites. Learners' adept use of email, voicemail, videomail, and direct audio/video can improve the spectrum of expertise available in their training. However, the types of task structures and interpersonal relationships required for effective usage of these media differ from those conventionally utilized in face-to-face interactions (Sproull, & Kiesler, 1991). Training learners to use technology-mediated communication effectively—whether over email or in artificial realities—involves more than simply mapping skills from face-to-face encounters into the virtual environment (Dunlop, & Kling, 1991).

As discussed earlier, situating education in virtual contexts similar to the environments in which learners' skills will be used helps their knowledge to transfer. When the material involved has psychosocial as well as intellectual dimensions, the design of authentic experiences to embed in artificial realities for education becomes more complex. In addition to physical and cognitive entities, instructional developers can include simulated beings (avatars and knowbots) in the virtual environment.

One example of such a training application involves software engineering education; students are trained in a technical process, code inspection, that is one stage of a formal methodology for software development (Stevens, 1989). Using hypermedia, Digital Video Interactive (DVI), and rule-based expert systems, the Advanced Learning Technologies Project at Carnegie Mellon University has created a virtual environment similar to a typical corporate setting. The trainee interacts with this artificial reality in the role of a just-hired software engineer still learning the profession. Through direct instruction and simulated experience, the student practices the process of formal code inspection.

The learner can access various rooms in the virtual software company, including an auditorium, library, office, training center, and conference facility. Machine-based agents (knowbots) that simulate people, such as a trainer and a librarian, facilitate the use of resources to learn about the code inspection process. Via specialized tools in the office, the student can prepare for a simulated code inspection, in which he or she can choose to play any of three roles out of the four roles possible in this formal software review process. For each inspection, a rule-based expert system utilizes DVI technology to construct knowbots that simulate the three roles not chosen by the learner. This knowledge-based system controls the topic of conversation; determines who should speak next; and models the personalities of the knowbots in the inspection meeting, altering their cognitive and affective perspective depending on what is happening.

The learner uses a menu-based natural language interface to interact with these simulated beings, who model behaviors typical in code inspection situations. The student not only can choose from a wide range of options of what to say, but can determine when to make remarks and can select the emotional inflection of his or her utterances, from a calm passive tone to an angry aggressive snarl. By mimicking the reactions likely from human participants in a real simulation, the knowbots provide the learner with a sense of the strengths and weaknesses of different intellectual/psychosocial strategies for that role in a code inspection.

Without using artificial realities and knowbots, this type of authentic experience is very difficult to simulate in classroom settings. Not only is the training environment dissimilar from the corporate context in which software development skills will be used; but also students do not know how to roleplay exemplary, typical, and problematic participants in code inspections. Through knowbots, the instructional designer can provide paradigmatic illustrations of how to handle a variety of situations, without the expense of having teams of human actors perform for each individual learner.

Distributing the intelligence incorporated into artificial realities among simulated beings raises challenging instructional design issues. The quality of what students learn is determined by the accuracy with which a knowledge-based system can simulate not only human reasoning, but also people's personalities and emotions. This is a very difficult task, necessitating knowledge acquisition about psychosocial dynamics as well as intellectual issues.

Another design issue posed by knowbots is whether students will profit more by interacting with a variety of partial machine-based intelligences (a collaborative learning/peer teaching perspective) or with a single, omniscient source of knowledge (an intelligent tutoring systems perspective). Students may empathize more readily with flawed simulated beings (e.g. a "good, but still learning" moderator for code inspections) than with an intelligent coach who has mastered every aspect of the educational domain. Programming several machine-based agents with highly focused skills is also easier than constructing a single, broad-based knowbot; knowledge-based systems work best for narrow, well-specified domains.

The simulated beings in artificial realities for education need not all be machine-based agents. Military and aerospace research projects have created telerobotic technologies that allow operators to control devices across distance (Fisher, 1990). These enable the creation of educational environments that provide a virtual link between remote learners and real world settings; for example, a student can manipulate industrial process technology without physically traveling to the factory (Uttal, 1989). An extension of teleoperations is telepresence, which allows people to interact across distance via avatars (computer-graphics representations), as opposed to direct video-links.

For example, SIMNET (Orlansky, & Thorp, 1991) is a training application that creates a virtual battlefield on which learners at remote sites can simultaneously operate military equipment. Complex data-objects that indicate changes in the state of each piece of equipment are exchanged via a telephone network interconnecting the training workstations ("dial-a-war"). The appearance and capabilities of graphics-based avatars representing military equipment alter second-by-second as the virtual battle evolves. Through this telepresence approach, a widely distributed group of personnel can engage in simulated real-time warfare without the necessity of gathering the participants at a single site to conduct combat.

Some of the representations in the SIMNET artificial reality are avatars controlled by human beings; others are knowbots whose actions are generated by a knowledge-based system under general human guidance. Similarly, the virtual world for code inspections described above could be redesigned to incorporate both avatars controlled by human agents and machine-based knowbots. For any given artificial reality for education, balancing the different types of beings interacting in the virtual environment (avatars, knowbots, video-links) is an interesting issue for instructional design research.

As users interact in virtual environments, intriguing interpersonal dynamics emerge that are quite different from typical real-world encounters. People participating in artificial realities often feel as if knowbots are real human beings, an illustration of the general principle that users tend to anthropomorphize any type of machine-based agent. Joseph Weitzenbaum's Eliza program, which simulates a Rogerian therapist, is an example of this tendency; some users type in responses to Eliza's prompts for hours, attributing human understandings to a pattern-matching natural language program that has no conception of people's emotions or behaviors.

As a complement to responding to knowbots as if they were human, participants in a virtual world interacting via avatars tend to treat each other as imaginary beings. An intriguing example of this phenomenon is documented in research on Lucasfilm's Habitat (Morningstar, & Farmer, 1991). Habitat was initially designed to be an on-line entertainment medium in which people could meet in a virtual environment to play adventure games. Users, however, extended the system into a full-fledged virtual community with a unique culture; rather than playing pre-scripted fantasy games, they focused on creating new lifestyles and utopian societies.

As an entertainment-oriented cyberspace, Habitat provided participants the opportunity to get married or divorced (without real-world repercussions), start businesses (without risking money), found religions (without real-world persecution), murder other's avatars (without moral qualms), and tailor the appearance of one's own avatar to assume a range of personal identities (e.g. movie star, dragon). Just as SIMNET enables virtual battles, Habitat and its successors empower users to create artificial societies. What people want from such societies that the real world cannot offer is magic, such as the gender-alteration machine (Change-o-matic) that was one of the most popular devices in the Habitat world.

Users learned more about their innermost needs and desires by participating in Habitat than they would have by spending an equivalent amount of time listening to psychology lectures. Similarly, social scientists are discovering more about utopias by studying Habitat's successors than they did by researching communes, which were too restricted by real-world considerations to meaningfully mirror people's visions of ideal communities. Giving users magical powers opens up learning in ways that trainers are just beginning to understand. As with any emerging medium, first traditional types of content are ported to the new channel; then alternative, unique forms of expression—like Habitat—are created to take advantage of expanded capabilities for communication and education.

Interactive, shared virtual worlds enable experiments to determine the leverage that collaborative learning provides in immersive artificial realities. How well students can use their usual face-to-face communication skills through the medium of a virtual reality interface via "avatars" (computer graphics representations of individuals) is an important topic for exploration. The types of functionalities that are important in empowering cooperative learning in a medium richer than electronic mail, but less direct than direct videolinks, also deserve investigation.

Unlike passive, linear media such as television, artificial realities center on immersion and collaboration. Education in the future may balance learning experiences in three types of environments: contemplation and introspection via books and symbolic media, immersion and collaboration in virtual communities populated by knowbots and avatars, and traditional student/instructor interaction in conventional teaching/learning settings.

Conclusion

Technology evolves in waves of innovation and consolidation. The advent of "motion pictures" about a century ago ushered in civilization's fourth medium, another dimension to communication beyond spoken language, written language, and still images. Later, new technologies appeared to embellish the capabilities of moving images: broadcast and narrowcast television, videotapes, videodiscs, multimedia, hypermedia. Now all of these are merging into a synthesis so far beyond its individual components that it constitutes a new medium: artificial realities.

Part of the educational implications of this medium center around its channel, which is rich and powerful enough to mimic the meta-medium in which we live, the real world. Other instructional implications come from the content that cybernauts are embedding into this channel: sensory and cognitive transducers, virtual communities made up of people's avatars and of machine-based knowbots. Together, channel and content form the message of this new, immersive medium, which is still too indistinct to fully comprehend, but appears both fascinating and frightening.

Any powerful information technology is a double-edged sword: a source of either propaganda or education. Through advances in information technology, virtual environments can now be created that seem intensely real to participants, yet may be false to the true nature of reality in the same way that fractally-generated mountain ranges are not valid depictions of physical topography and geology. Artificial realities, virtual communities, and knowbots are emerging technologies whose fusion with multimedia/hypermedia has enormous potential to improve education. However, teachers and learners must recognize that these instructional vehicles carry intrinsic content that can empower or subvert the goals of an educational experience. Careful research is needed to understand how to optimize the design and utilization of virtual worlds for education.

References

Brooks, Jr., F.P. (1988). Grasping reality through illusion: Interactive graphics serving science. *CHI '88 Proceedings* (pp. 1-11). Reading, MA: Addison-Wesley.

Brown, J.S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher 18*(1), 32-42.

Dede, C., & Palumbo, D. (1991). Implications of Hypermedia for Cognition and Communication. *Impact Assessment Bulletin 9*, 1-2 (Summer), 15-28.

Dunlop, C., & Kling, R. (1991). Social relationships in electronic communities. In C. Dunlop & R. Kling (Eds.), *Computerization and controversy: Value conflicts and social choices* (pp. 322-330). New York: Academic Press.

ERIC

Fisher, S. (1990). Virtual environments, personal simulation, & telepresence. *Multimedia review: The journal of multimedia computing* 1(2), 34-44.

Gibson, W. (1984). Neuromancer. New York: Ace Books.

Greif, I. (Ed.). (1988). *Computer-supported cooperative work: A book of readings.* San Mateo, CA: Morgan Kaufmann.

Malone, T.W., & Lepper, M.R. (1985). Making learning fun: A taxonomy of intrinsic motivations for learning. In R.E. Snow & M.J. Farr (Eds.), *Aptitude, learning, and instruction: III. Conative and affective process analysis* (pp. 176-189). Hillsdale, NJ: Lawrence Erlbaum.

Minsky, M., Ouh-Young, M., Steele, O., & Brooks, F.P. Jr. (1990). Feeling and seeing: Issues in force display. *Computer graphics: Proceedings of the 1990 symposium on interactive 3-D graphics* 24(2), 235-244.

Morningstar, C., & Farmer, F.R. (1991). The lessons of Lucasfilm's Habitat. In M. Benedikt (Ed.), *Cyberspace: First steps* (pp. 273-302). Cambridge, MA: MIT Press.

Orlansky, J., & Thorp, J. (1991). SIMNET — An engagement training system for tactical warfare. *Journal of Defense Research* 20(2), 774-783.

Resnick, L. (1987). Learning in school and out. *Educational Researcher 16* (10), 13-20.

Robertson, G., Card, S., & Mackinlay, J. (1991). The information visualizer. *Proceedings of CHI 91* (pp. 181-188). New York: Association for Computing Machinery.

Schrage, M. (1990). *Shared Minds: The New Technologies of Collaboration.* New York: Random.

Smith, R.B. (1987). Experiences with the alternate reality kit: An example of the tension Between literalism and magic. *Proceedings of CHI+GI 1987* (pp. 324-333). New York: Association for Computing Machinery.

Sproull, L., & Kiesler, S. (1991). Computers, networks, and work. *Scientific American* 265(3), 116-127.

Stevens, S. (1989). Intelligent interactive video simulation of a code inspection. *Communications of the ACM* 32(7), 832-843.

Tufte, E.R. (1990). *Envisioning information.* Chesire, CN: Graphics Press.

Uttal, W.R. (1989). Teleoperators. *Scientific American* 261(6), 124-129.

Walker, J. (1990). Through the looking glass. In B. Laurel (Ed.), *The art of computer-human interface design* (pp. 213-245). Menlo Park, CA: Addison-Wesley.

ERIC

"Computer Supported Cooperative Learning : a real-time multimedia approach"

A.C. Derycke*, P. Croisy*, P. Vilers**
*Laboratoire TRIGONE - Institut CUEEP and
Laboratoire d'Informatique Fondamentale LIFL
Bât. C6 - Cité Scientifique - 59655 VILLENEUVE D'ASCQ Cédex - FRANCE
e_mail : derycke@trigone.citililie.fr
Phone: +33.20.43.49.21 - Fax : +33.20.43.60
**IUFM -Lyon France - 5 rue Anselme - 69004 LYON

I - Introduction

The new technologies of communication have a rapid evolution which is due to two inter-dependant progress :

- the rapid progress of data networks ;
- the integration of different media and interactive devices in the same workstation.

This technological progress has given opportunity to develop new user-centred applications. This is of course the case of hypermedia for the access of multimedia database (Mulhauser, 1991). But it is also all the various kind of remote cooperation or collaboration such as Tele-medicine, Tele-Expertise and of course Tele-teaching.

The domain of the Computer Media Communication is rapidly changing to reflect the aforementioned evolutions. The Computer Supported Cooperative Work takes benefit of these new possibilities. Traditional group communication systems such as audio or video conferencing take new ways with their association with the computer which can give a rapid spread of their uses (Watabe, Murakami, Ishikowa, & Kamae, 1985).

Our interest is to apply this potential to the education field. In effect, the important needs for education, either initial for kids or continuing education for adults, have produced new learning methods and systems. Distant learning or Open Learning has given a new development with the help of multimedia technologies and tools.

However, the focus is too often put on production of multimedia documents for the courses (the authoring process) and their electronic deliveries (Multimedia database). In fact it is also important that the learners, especially those who are insulated in their homes or enterprises, have access to "social resources" : i.e. tutors, teachers, advisors or others learners.

So we have developed some tools to support cooperative learning which are analog to those implemented in the field of CSCW or groupware. For this purpose we have designed an open platform which supports both asynchronous or synchronous communication (Derycke, Viéville, & Vilers, 1990). In this paper we will present only the synchronous aspect which deals with multimedia and extension to broadband [1].

II - A general architecture for supported distant cooperative learning in real time

II.1 - An overview

Our architecture was severely constrained by the educational context of our applications. The first constraint is the cost of the learner's workstation and the network access. The second is the disseminated nature of the users : they are spread in various location which could be home, small or medium enterprises, big companies or learning resource centres. This configuration is quite different that those encounter is the field of Office Automation for CSCW of groupware (Crowley, Milazzo, Baker, Fordsick, & Tomlinson, 1990). Effectively the learners are not members of the institution which delivers the distance teaching. They are temporary users of the systems and they can have access from several different locations and workstations. Those reasons, combined to the necessity to control the access to the shared resources, lead to choose a relatively centralised architecture which can be attenuated by a client/server relationship.

The figure 1 gives an overview of the hardware platform and network architecture. The two channels of the basic ISDN (Integrated Services Data Network) access is used one for voice (7 KHZ quality) and one for the data (64 Kbit/s available). The learner workstations have access simultaneously to the conference server and to an audio server which controls the audiobridge. So it is possible to exchange multimedia documents to share special multi-user design tools and to have verbal conversation at the same time.

It must be noted that at this stage, the active media (voice), is not fully integrated in the computer system because the sound is not stored in the different memories (it is not a store and forward system).

[1]A part of this work is now done in the framework of the CO-LEARN project, one of the DELTA EEC program fund.

ERIC

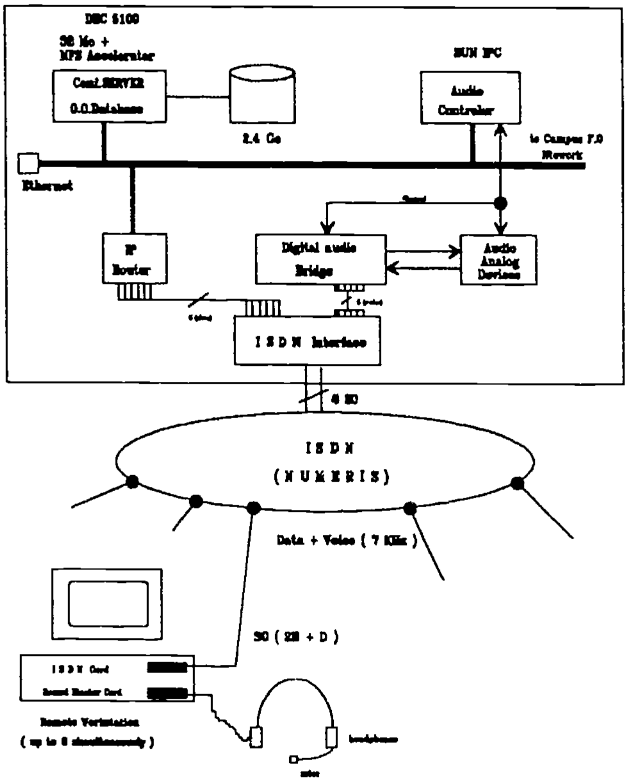


Figure 1 : Real Time Multimedia Conferencing system based on narrow-band ISDN

II.2 - Software architecture

All the previous considerations have pushed to choose a hybrid architecture based on client/server model : each user connects his workstation to the server which establishes inter-users connections, manages the audio bridge, and guarantees the coherence at the applications levels. The fully replicated architecture was rejected because of potential problems of maintaining identical state across all copies of an application. In fact this is more severe with than in LAN because the possibility of short cut of the physical connection during a session and less low reliability at the packet level.

The software for the workstation and server has been developed in an object oriented environment in order to favour an easier and incremental prototyping : the Smalltalk-80 environment (Goldberg, & Robson, 1983).

So the first step of our work was to design and implement communication and co-operation mechanisms in the Smalltalk environment (Viéville, Derycke, & Vilers, 1990). We develop a mechanism which offers the possibility to access to a remote object : proxy mechanism. It is completely transparent at the user level. We use the same syntax to access to the remote object.

For the user, the system is viewed as a multi-window screen :

- a public window which is the interface with the dedicated co-operative application (the groupware). The central concept is the "WYSIWIS" (What You See Is What I See) with still the relaxation of some constraints especially the spatial constraints (possibility to manage the local view of the shared objects) (Stefik, Foster, Bobrow, Kahn, Lanning, & Suchman, 1987) ;
- a window relative to the global management of the teleconferencing system : the interface allows the possibility to establish or to close a connection with such and such conference, it gives list of active participants and a feedback of their activity and roles, and it allows access to the cooperative application via a control mechanism base on "floor algorithm" ;
- private window with the possibility in some case to exchange data with the public one by a cut-and-paste operation. We use also a more sophisticated mechanism to communication between this two worlds (like the DDE or OLE in Windows 3.1).

We have also implemented an object oriented toolbox which can offer in the shared cooperative application some new functionality such as telepointer or a vote support which automatically collect the results, to help the users to collaborate.

II.3 - Broadband extension

We use the potential of new hardware additional boards for PC/Computer under Window 3.1 or for Unix Workstation under XWindow. These Boards allow the incrustation of live video in one window of the workstation and its control : size, overlay of text or graphic, storage of captured image. The inputs are compatible with the NTSC format (640 X 480 pixels) in North America and with PAL (768 X 576 pixels) in Europe. The sources can be a camera, a tape video recorder or a videodisk.

Of course our intent is to put this Video Window in the WYSIWIS mode with a shared control of the source and some kind of Telepointer. This appears to be important for some of our learning strategies because we already use video to show and discuss various real situations recorded in the factory for example (see the following chapter).

Due to the high cost of access to broadband network at a long distance, we have restricted this broadband extension to a same building with some remote access via a fibre optic network to some remote places located of the same campus.

The figure 2 gives an overview of the architecture of the network we have chosen. In this phase it is relatively conservative because the video sources and transport are still analogic until the input of the workstation. It is the same for the High fidelity Sound. So there is no really an integration at the network level but this integration is realised at the workstation level.

ERIC

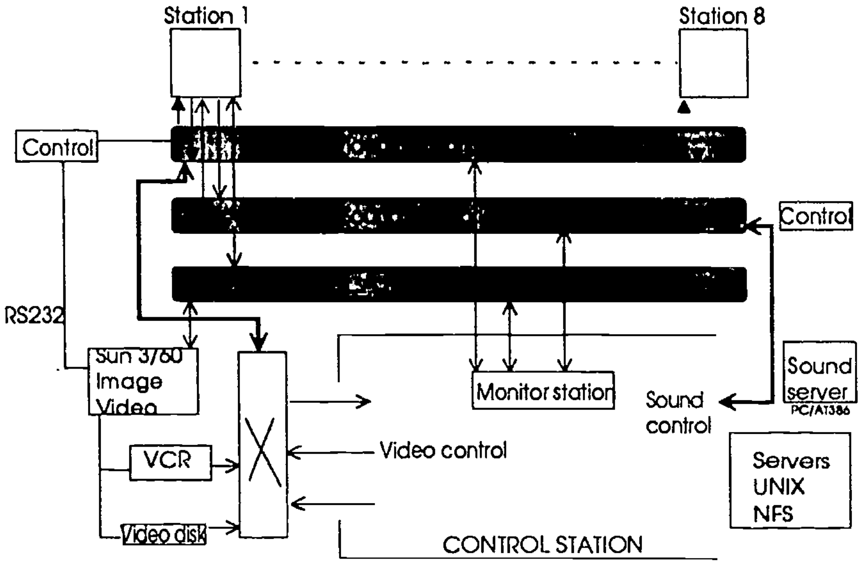


Figure 2 : Broadband architecture

Now the first phase of our implementation is already done : the hardware and network are fully operational. The second phase we are doing is relative to the multi-user control of the video-source in a client/server architecture and the introduction in the Video Network of various special effect hardware (video sources composition, overlay of sources...) which will be also computer controlled.

III - A real time conferencing application devoted to the manufacturing management learning

We have chosen to design and implement cooperative application in the learning of the management of the manufacturing process in the factory (MRP or Kanban methods for example). This methods are not classically taught. They are learned by an intensive co-operative simulation of the factory (similar to some games in the field of economy education). This field was chosen because not only its direct interest in education and our previous experience but also because it is a rich and multifaceted application.

In effect during the evolution of the simulation, which can last more than twenty hours if several planning methods are studied (in two hours sessions), various situations, activities, roles are found, and so different types of co-operation (steps with high and low degrees of co-operation).

BEST COPY AVAILABLE

- for example in the first step which is the general presentation of the simulation. Group discussion arises around common documents presented in a strict WYSIWIS manner. But each learner has also the possibility to read documents in his private windows. The role of the instructor/expert is to start the discussion and conduct it. The control is principally verbal. The instructor sees the same public window that the learners ;
- in the second step, there is a cooperative activity to establish the general manufacturing planning (number of pieces of hardware produced by the different workshops, characteristics of each batch, previous stocks...). The attention is focused on shared tools (groupware) such as a cooperative spreadsheed with possibilities of conditional submission of values, vote to organise the consensus ;
- the third step is the simulation phase. Each learner conducts a subpart of the plan (a workshop). The public window gives in effect different points of view depending of the subpart considered. The instructor has different tools to control the simulation evolution, to introduce some events such as failure in some machines or other problems. He has the possibility to stop the simulation to open a group discussion about different topics. In the third step the coordination is less important. Each workshop (workstation) communicates with the others by shared data : stock values, time... ;
- the last step is the analysis of the finished simulation. It uses tools analogous to those of the step one.

Of course, the computer which assists distributed simulation allows storage of different steps in order to analyse them after the end of the simulation. It gives also the possibilities to keep a personal memory of different documents or snapshot of some phase of the simulation.

IV - Comparison with other works and future extensions

IV. 1 - Comparison with other works

In the field of Education there is a few approaches of co-operative learning using real time multimedia computer conferencing. Shared-ARK is a distributed systems to investigate cooperative problem solving in the field of physics courses (Smith, O'Shea, O'Malley, Scanlon, & Taylor 1991). It is a very interesting experience. However this system is not an open system to support implementation of various pedagogical activities. And until now it is only for point-to-point interaction such a tele-tutoring of a learner. Our system is more generic and it is oriented to small group learning (less than 10 learners) spread over various distant sites

Another project is relative to distributed group problem solving using of cooperative game (Dekoven, & Radhakrishnan, 1990) but its goals in more an evaluation of how a team of human resolves problem (for design of future distributed expert systems) rather than the learning process and its effectiveness.

IV.2 · Future extensions of our work

The work will be followed in two directions :

- the first in the framework of the DELTA CO-LEARN project. In an international consortium our goal is to integrate the real time multimedia conference system in a more general teleteaching system based of narrow band ISDN. For this purpose we are modifying the conference manager in order to be tailored to various needs : learning strategies and scenarios. An interesting challenge is the integration of the asynchronous mode of communication (a general Computer Mediated Conference). For this purpose we are constraint to do a more advanced work on the man machine interface and the general metaphor which is underlying. The ROOMS metaphor we are investigating seems to be powerful for this purpose (Henderson, & Card, 1986) ;

- the second direction is relative to the multimedia integration at the user's level and in the Computer Supported Collaborative Learning perspectives the Work is done at two levels :

* the hardware and network level : the major problem deals the synchronisation of the various media sources. We conduct an intensive research on the sound aspect than, in most of the CSCW systems, is often neglected ;
* at the Multimedia database level : we are using an object oriented database, Gemstone, to store some of the multimedia messages (in the asynchronous mode) and to give more functionality to the video or audio server.

Conclusion

The potential of multimedia for education is still to be explored. Two ways are offered. The first one is to design tools to produce very sophisticated Hypermedia learning documents. This is very interesting because it maintains all the best features of Computer Assisted Instruction, especially the possibility of selfspace learning, and gives a more flexible possibility to support various learning strategies.

The second way is to use multimedia systems to support collaborative learning in the same location or remote place. This is this way we have chosen to investigate and for this purpose we have implemented an open architecture which enables us to support various kind of Groupware specialised for education. We are trying to give a balance between the individual use of video media and its collective uses. This will easier in the next years with the fast growing of broadband network.

Références

Crowley, T., Milazzo, P., Baker, E., Fordsick, H., & Tomlinson, R. (1990). MMconf : an infrastructure for building shared multimedia applications. *Proceeding of the CSCW'90 ACM conference*, Los Angeles CA, 329-342.

ERIC

Dekoven, C., & Radhakrishnan, T. (1990). An experiment indistributed group problem solving. *Multi-user Interfaces and Applications*, S. Gibbs and A.A. Verijn-Stuard (Eds), North-Holland, 61-76.

Derycke, A., Viéville, C., & Vilers, P. (1990). Cooperation and Communication in Open Learning : The CoCoNut Project. *Fifth World Conference on Computer in Education*, IFIP, Sidney, Australia, 957-962.

Goldberg, A., & Robson, D. (1983). Smaltalk 80 : The language and its implementation. Addison Wesley.

Henderson, D.A., & Card, S.K. (1986). Rooms the use of multiple virtual workspaces to reduce space contention in a window base graphical user interface. *ACM transaction on Graphics*, 5(3), 211-243.

Mulhauser, M. (1991). Computer based learning with distributed Multimedia systems. *Human aspects in Computing : design and use of interactive systems and Information management*, Bullinger H.J. (eds), Elsevier science Pub, 953-958.

Smith, R.B., O'Shea, T., O'Malley, C., Scanlon, E., & Taylor, J. (1991). Preliminary experiments with a distributed, multimedia, problem solving environment. *Studies in Computer Supported Cooperative Work*, J.M. Bowers and S.D. Benford (Eds), North-Holland, 31-48.

Stefik, M., Foster, G., Bobrow, D.G., Kahn, K., Lanning, S., & Suchman, L. (1987). Beyond the chalkboard : Computer support for collaboration and problem solving in Meeting. *Communications of ACM*, 30(1), 32-47.

Viéville, C., Derycke, A., & Vilers, P. (1990). Architecture of collaborative system using Smalltalk and Unix. *European Unix User Conference Proceeding*, Munich, RFA, 89-98.

Watabe, K., Murakami, S.I., Ishikowa, H., & Kamae, T. (1985). Audio and visually augmented teleconferencing. *Proceeding of IEEE*, 73(4), 656-670.

Watabe, K., Sakata, S., Maeno, K., Fukuoka, H., & Ohmori, T. (1990). Distributed Multiparty desktop Conferencing System : Mermaid. *Proceeding of CSCW'90 ACM conference*, Los Angeles, 27-38.

Concept mapping as a mind tool for exploratory learning

Sjoerd de Vries and Piet Kommers
Centre for Applied Research in Education and Department of Education, University of Twente, P.O. Box 217, 7500 AE, Enschede, The Netherlands.

Introduction and background

In 1987 the Dutch Minister of Education and Science started the Experimental Schools (ES) Project. The aim of the ES Project is to investigate various ways of realising the integration process of information technology in schools. At the Centre for Applied Research in Education at the University of Twente five projects are conducted on behalf of the ES Project. Two pilot schools for non-vocational Secondary Education in Enschede are taking part in these projects. This article is based on the findings of one of the projects, that is the TextVision project.

The TextVision project aims at the use of hypermedia programs in education, in particular at the possibilities of supporting the learner in retrieving and processing information from knowledge corpora. In the project the principle of concept mapping (Anderson, 1977) is combined with the facilities that are provided by a hypermedia program.

Knowledge corpora and education

The availability of large quantities of information by the excessive use of computer systems requires an education in which students are learned how to get the right information out of these resources. Digitised information resources are here being referred to as knowledge corpora.

Use of knowledge corpora in education

The use of knowledge corpora by means of the computer in education is very important. More and more it is required that people handle large knowledge corpora. Learners need to practise skills that enable them to handle these corpora. The computer has the potential to provide support with specific tools which can not easily be provided in another way.

The TextVision project aims at the use of hypermedia programs in education. To make use of hypermedia programs in education Hypermedia Oriented Courseware (HOC) is needed. The purpose of the TextVision project is to realise the use of HOC in non-vocational education and to explore the added value of the use of HOC for instruction and in particular to the possibilities to support the learner in retrieving and processing information from knowledge corpora.

Hypermedia programs

The use of Hypertext and HOC in education offers a wide range of opportunities for the teacher and the student, but implies also various problems (Kommers & de Vries, 1992). This type of courseware provides inquiry oriented instruction that is based on the view of students: '... as active constructors of knowledge, of knowledge as open and evolving, of academic learning as exiting and vital, and of teaching as a stimulus to curiosity and a model of inquiry' (Cohen, 1988). Hypermedia programs are used in education to interact with knowledge corpora and to utilise the corpora in a way that fits the curriculum and the classroom situation. The learning goal of retrieving, interpreting and creatively using information from knowledge corpora is of growing importance in education (see also Cummins & Sayers), but the feasibility of such learning goals in the reality of school environments is a problem. Conventional methods of information management are not sufficient for organising and accessing a rapidly expanding mass of data (Raker, 1989; Layman & Hall, 1991).

Hypermedia programs provides at least three facilities that support the use of hypermedia programs in education. Hypermedia programs offer the user options to inquire and explore knowledge corpora in an interactive way. Hypermedia programs also have the potential to organise and construct knowledge by supplying add-, delete- and edit- functions. Finally, they provide built-in options and features to support the user in processing the information that is derived from the knowledge corpora, for example, by means of instructional questions, or special delivery methods of information.

Therefore, using hypermedia programs in education is expected to provide an added value of instruction. The expected added value of instruction is based on three expectations of the use of hypermedia programs in education. (1) Hypermedia programs let teachers and students interact with large multimedia storage systems in an open and flexible way. Open and flexible access means that the user is in control and decides what to do. This implies that the user has to develop his own search methods. The computer can support the user in this process of information retrieval. (2) Hypermedia programs offer users the possibility to construct and adapt information in a multimedia storage system in an individual manner. This enhances the involvement of the user in structuring the information and will lead to an increase of his motivation. (3) Hypermedia programs support the user in processing the information by providing built-in options and features. This contributes to the effectiveness of the use of a multimedia storage system in education and is of benefit to the learners. They learn how to use hypermedia programs for retrieving information.

However, there are at least four problems that are related to the use of hypermedia programs in education. The first two problems are related to the usability of such programs. The usability is the extent to which an end-user is able to carry out required tasks successfully, and without difficulty, using the computer application system (Ravden & Johnson, 1989). The first two problems are related to the quality of learning. (1) The usability of hypermedia programs is not always considered as satisfactory. A problem related to the usability is for example the orientation and disorientation in the multimedia storage system (Locatis, Letourneau & Banvard, 1989; Tripp & Roby, 1990; Kinzie & Berdell, 1990). (2) The possibility of adapting the information in a multimedia storage system by the users leads to problems in the maintenance of the system and in sustaining the quality of the system (Kaplan & Maarek, 1990). (3) The students as the potential users may lack the cognitive skills and dispositions required to take advantage of the functions offered by hypermedia programs. In (Gay, Trumbull & Mazur, 1991) it is shown that: some learners are in the habit of accepting information passively or of consistently seeking external assistance, and that many individuals have trouble adjusting to self-directed learning. (4) The effective use of large multimedia storage systems demands new information-seeking strategies (Layman & Hall, 1991; Gay, Trumbull & Mazur, 1991; Marchionini, 1989).

Concept mapping and TextVision

Concept mapping is the technique based on the notion that human knowledge is primarily organised in a schematic format, and students should be motivated to externalise prior and new knowledge in a spatial way so that it can easily be observed for internal consistency and global layout (Anderson, 1977). Besides the meta-cognitive functioning of concept mapping tools, it has proved to be of value for the navigation complexity in extensive hypertext networks. A concept map, that is the schematic representation of links between concepts as they evolve from descriptive paragraphs in hypertext, is much more effective to decide a certain route through the information space, compared with simply clicking in the subsequent hot spots. The combination between meta-cognition (making explicit which concepts belong to a certain knowledge domain) and the concept map as a compact orientation device for consulting new information offer possibilities to do in-depth experiments in learning settings.

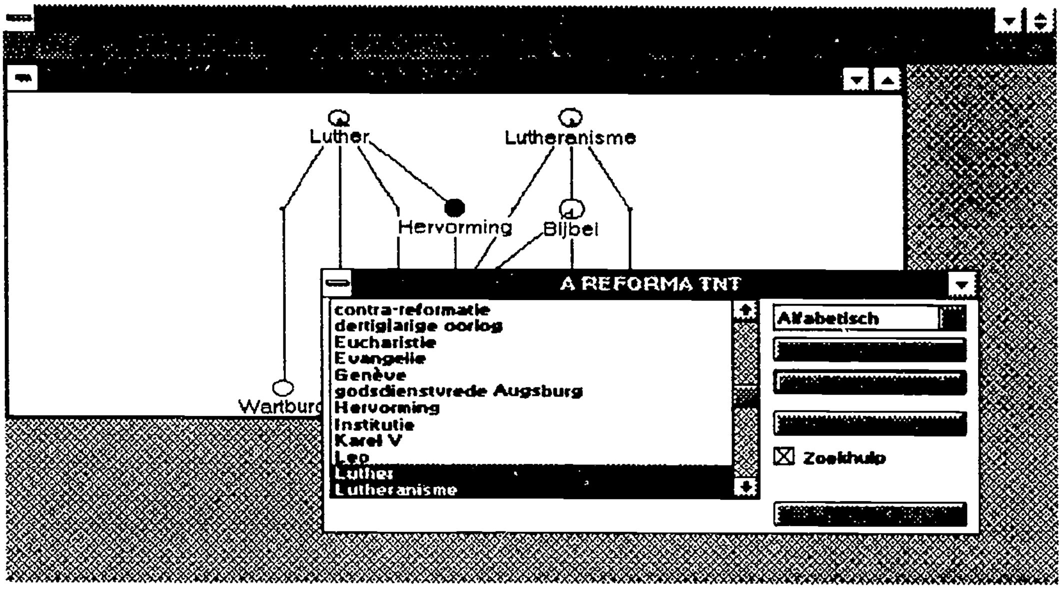


Figure 1 Two possible ways of navigating when using a concept base in TextVision-2.

In this project the principle of concept mapping is combined with the facilities of a hypertext program. The multimedia program TextVision-2 gives access to a knowledge corpus that involves a so called concept base. A concept base consists of concepts or nodes. Each concept is attached to a text file and has links with other concepts. Links between concepts can be derived from the mark-up language (S)GML in an original resource like encyclopaedia, newspaper or technical reference manual. However the (second) author like the teacher, should be able to revise previously assigned relations, and he should be able to rewrite the text. The concepts and the texts we use are extracted from a digitised encyclopaedia and form a base of 140-160 concepts that are linked by seven different types of links. The concept base has been tailored to the target group of learners (13-14 years old). The quality of the concept base is very important. Putting concepts, links and texts into a concept base is easily done, but making the information meaningful and consistent is a problem.

TextVision-2 supports the construction, adaptation and augmentation of the information in the concept base. It provides views of the concept base by means of concept maps. These maps can also be constructed, edited, combined, stored and retrieved by the user. In other words, the user can construct his own views based on the concept base.

In using the concept base for information retrieval, a user may browse sequentially through the concept base or he may use a concept map to explore concepts by following the links in-between (Figure 1).

BEST COPY AVAILABLE

In an educational setting, the content of a concept base has been assembled by a teacher or a domain specialist. The learner is the actual end-user, but his activities with the TextVision-2 database is restricted to read, select and zooming operations. TextVision-2 enables learners to define selective views on a medium-size (100-200 nodes) network.

The TextVision-2 program runs under Windows and has an interface with the following characteristics: the text and graphic level is easily accessible, switching can be done quickly and easily; texts of different concepts can be presented at the same time; the importance of a concept can be visualised.

TextVision-2 and education

In the TextVision-project the multimedia program TextVision-2 has been developed and used in the two pilot schools. The goal was to find out whether there are aspects of added value for instruction in relation with the use of hypermedia programs in education. These aspects are specified earlier in terms of the positive expectations and expected problems of the use of hypermedia programs in education. The development and testing of the TextVision-2 program took place in co-operation with three teachers History of the two Experimental schools. In the school years 1990-1991 and 1991-1992 the program was experimented with during courses of three to six lessons in second classes. Learners used TextVision-2 to search for information necessary for writing a paper. We participated in the lessons and observed the teacher and the learners working with TextVision-2. Afterwards we gave six other teachers who all had participated before in the TextVision project an introduction in the use of TextVision-2 and in the way the program was used in the lessons. All nine teachers answered a questionnaire about the added value for education and afterwards the three teachers who used TextVision-2 in the lessons were interviewed about the use of TextVision-2 and the aspects of added value for instruction. The learners (n=110) also answered a questionnaire about the use of TextVision-2 and the added value of the use of the courses. We analysed the data in a qualitative and quantitative way and based on the results we try to answer whether there are aspects of added value for instruction.

Lessons designed with use of TextVision-2

Two courses were developed by making use of TextVision-2 and a concept base.

Each course consisted of learning goals and instructions for the learner. A description of the course was presented on paper to the learner. In a short description of these two courses an impression is given of the way TextVision-2 and the concept base are used in the classroom. We have used the courses in sum total five second classes at the pilot schools, the age of the learners was about 14 years.

The first course -- named 'Getting Access to large Concept maps 1 - (GAC-1) -- was aimed at the construction of an essay of three pages. The learners had to collect information about a subject, they had to organise the information, and they had to write an essay. Information could be found in the study book, the concept base, and the library. The course consisted of six lessons and the students spent an average of one lesson spread over the six lessons on TextVision-2. TextVision-2 was used as a tool for searching and retrieving information.

The second course -- named 'Getting Access to large Concept maps 2 - (GAC-2) -- was aimed at the construction of concept schemes that needed to present a well-defined subject. The learners constructed concept schemes and they had to explain the relations in detail. They were able to derive information out of a concept base, but they had to edit the concept schemes. A group of learners consisted of two/three learners. They had to agree in the group about the resulting concept scheme. TextVision-2 was used as a tool for searching and retrieving information, and for the construction of concept schemes.

TextVision-2 in the classroom

In the schoolyear 1990-1991 GAC-1 was introduced in a second year History class. The learners (n=75) had already attended lessons in the subject matter of the course and they had to complete a test after the course. The learners had experience with the use of computers, and some of them were familiar with the principles of TextVision-2, because they had worked with the program TextVision-1 in the first class (Kommers & de Vries, 1992). We asked the learners by means of a questionnaire about their experiences. The meaning of the teachers were asked in an interview.

In the school-year 1991-1992 we have introduced GAC-2 at one experimental school in two classes (n=35), two teachers were involved. The target group had the same characteristics as the target group of the first introduction. The tasks of the learners were different. The learners had to construct concept schemes and they explained in detail all the relations. They derived information from the concept base, and constructed concept schemes. The learners worked in pairs and they had to agree in the group about the resulting concept scheme. The learners completed the same questionnaire as the one that was used in the first introduction, the results between the GAC-1 and GAC-2 were compared.

Results: added value for education?

The meaning of the teachers about added value for education of the use of TextVision in the classroom is presented in the figure below (Figure 2). The aspects of added value are (1) the possibilities of realising the principles of TextVision-2, (2) the feasibility of the learner goals, (3) the possibilities for new instructional settings, (4) possibilities to design lessons, and (5) the use of TextVision-2 in the classroom. The meaning is expressed in a general opinion, and in two aspects: difficult/time consuming, and meaningful/effective.

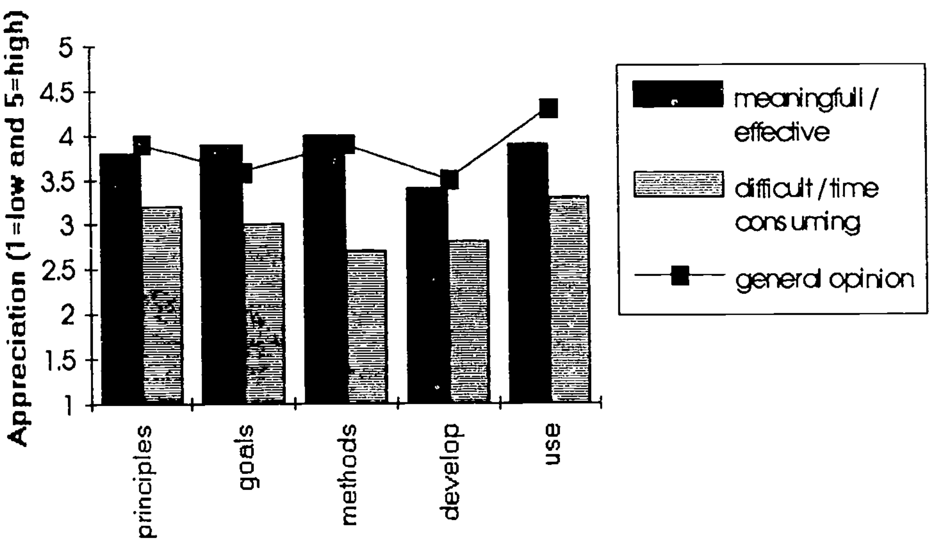


Figure 2 The opinion of teachers (n=9) about aspects of added value.

The use of TextVision-2 in the classroom got the highest score (most positive). The teachers thought about the use of TextVision-2 as meaningful, effective and not difficult and time consuming. This was interesting because the lessons with TextVision-2 in the computerroom were very different from normal lessons, and mostly very energetic. The lowest score got the aspect of designing lessons in relation with their own instruction environment. Although it was quite easy to make the instructions, and the concept base could be considered as given, it turned out that designing lessons was not considered as meaningful and effective.

The general opinion of the teachers was in general high and three out of five times the highest. Also is clear that the teachers are more positive about the meaningfulness/effectivity than about the difficulties and the time consuming aspects. The overall opinion of teachers is more influenced by their meaning about meaningfulness and effectiveness then difficulties and problems related to time. The possibility of new instructional settings is considered as the most meaningful and effective, but also as the most difficult and time consuming. The instructional setting can be characterised by: learners have a lot of freedom in performing the tasks, and teachers are not the presenters of the content matter but more or less guiding the students. They are not used to the change in role and behaviour of themselves and the learners. In interviews the teachers are asked about their meanings about learning results from the courses. Important was that the learner goals were on the level of analysing information and practising study methods and that they could not be reached in one course of six lessons. An important constraint was that the learner goals were not tested by central tests or exams and a part of the teachers admitted that they preferred that other teachers spent their time on reaching the just mentioned learner goals.

Table 1 The opinion of the learners (110) about positive and negative aspects of lessons with TextVision.

Opinion about the use:	60 % were positive
Two most mentioned positive reasons :	1) Sociable environment (66.4 %) 2) Variation in lessons (60.0 %)
Two most mentioned negative reasons:	1) Little explanation (34.5 %) 2) Learn more in normal lessons (31.8 %)

The learners were asked about their appreciation of the lessons with TextVision-2. The findings are presented in table 1. 60 % of the learners (110) were positive about the use of TextVision-2. The learners had to indicate two positive and two negative aspects of the use. It was interesting that more than 60 % of the learners did not indicate two negative aspects. The two most mentioned positive aspects were the sociable environment and the variation in lessons. The learners indicated for example nothing about 'interesting new learner goals'. Two most mentioned negative aspects were too little explanation and I learn more in normal lessons.

In the next figure (Figure 3) the appreciation by the learners of several aspects of the use of TextVision-2 is shown. A difference is made between the first (1990-1991) and the second (1991-1992) course.

Usability of TextVision-2 is an important aspect because the time span learning a program should be minimised. The usability was the second time considerably higher. The re-design of TextVision-2 was very easy to use. Information retrieval was an important aspect in the two courses and the second time special functions for information retrieval were added to the program. But, the learners still did not appreciate the aspect of information retrieval. The feasibility of the learner goals was according to the learners in the second course adequate. The motivation of the learners did not change over the two years and was satisfactory.

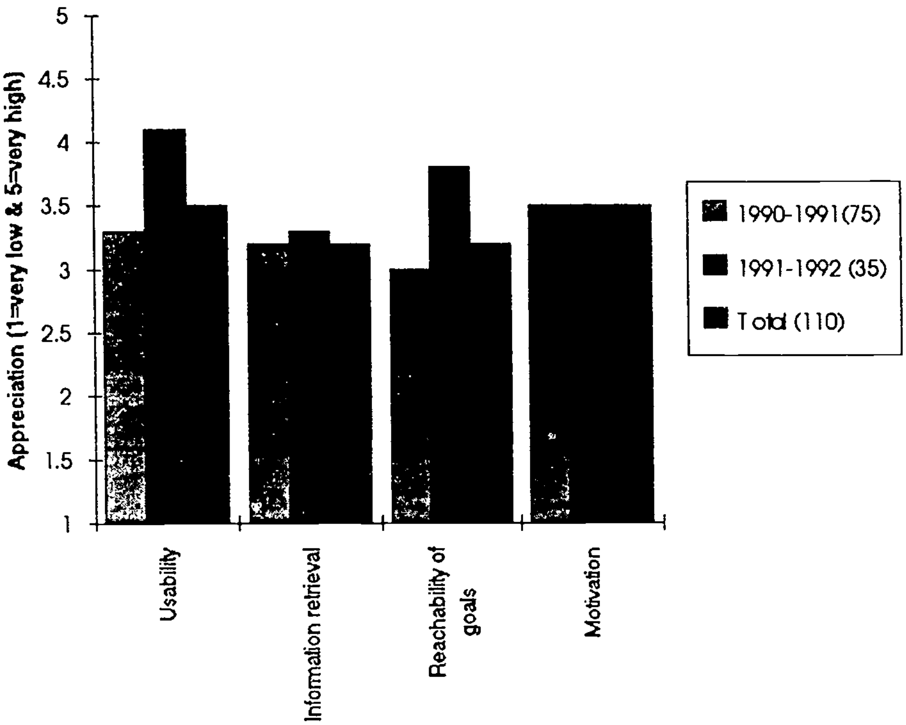


Figure 3 *The opinion of learners about the usability, possibilities of information retrieval, the feasibility of the learner goals and the motivation during the computer lessons.*

Conclusions and discussion

The study technique concept mapping can be used to guide learners when using knowledge corpora in education. TextVision-2 incorporates concept mapping and learners can explore and learn from concept maps by means of the concept mapping procedure. The results indicate that the appreciation of the added value for education according to the teachers and learners is very high. The presence and use of knowledge corpora in the home and work environment is growing and education needs to give significant attention to learning how to deal with them. Teachers consider the attention as time consuming and difficult related to their normal lessons. It is necessary that schools and governments emphasise the necessity of working with knowledge corpora. Learners are not used to student steering as is required by working with knowledge corpora. The learners' remark that they did not receive enough explanation and that they learn more in normal lessons, is possibly influenced by the little experience with student steering. This problem can only be solved by regularly practising (guided) student steering.

References

Anderson, R.C., (1977). The notion of schemata and the educational enterprise. In: R.C. Anderson, R.J. Spiro & W.E. Montague (eds). *Schooling and the aquisition of knowledge.* Hillsdale, NJ: Lawrence Erlbaum.

Cohen, D.K. (1988). Educational technology and school organization. In R.S. Nickerson, & P.P. Zodhiatis (Eds.), *Technology in education: looking toward 2020.* Hillsdale, NJ: Lawrence Erlbaum.

Cummins J. & Sayers, D. (1990). Education 2001: Learning networks and educational reform. *Computers in the schools, 7* (1/2), 1-30.

Gay, G., Trumbull, D., & Mazur, J. (1991). Designing and testing navigational strategies and guidance tools for a hypermedia program. *Journal of educational Computing Research, 7*(2), 189-202.

Kaplan, S.M., & Maarek, Y.S. (1990). Incremental maintenance of semantic links in dynamically changing hypertext systems. *Interacting with computers, 2*(3), 337-366.

Kinzie, M.B. & Berdel, R.L., (1990). Design and use of hypermedia systems. *Educational Technology Research and Development, 38*(3), 61-68.

Kommers, P.A.M., de Vries, S.A. (1992). TextVision and the Visualization of Knowledge. School-Based Evaluation of its acceptance at two levels of schooling. In: Kommers, P.A.M., Jonassen, D.H, Mayes, J.T. (eds). *Cognitive tools for learning.* Berlin Heidelberg: Springer-Verlag.

Layman, J. & Hall, W., (1991). Applications of hypermedia in education. *Computers and Education, 16*(1), 113-120.

Locatis, C., Letourneau, G., & Banvard, R. (1989). Hypermedia and instruction. *Educational Technology Research and Development, 37*(4), 65-77.

Marchionini, G., (1989). Information seeking in electronic encyclopedias. *Machine-Mediated Learning, 3*(3), 211-226.

Raker, E.J., (1989). Hypermedia: new technology tool for educators. *The Computing Teacher, 17*(1), 18-19.

Ravden, S.J., & Johnson, G.I. (1989). *Evaluating usability of human-computer interfaces: a practical method.* Chichester: Ellis Horwood.

Tripp, S.D. & Roby, W., (1990). Orientation and disorientation in a hypertext lexicon. *Journal of Computer-Based Instruction, 17*(4), 120-124.

A Survey of Applications of CSCW Including Some in Educational Settings

PRASUN DEWAN
Department of Computer Sciences
Purdue University, W. Lafayette, IN 47907, USA

CSCW (Computer Supported Cooperative Work) is an idea that was demonstrated as early as in the 1960s by Engelbart's pioneering work on the On-Line System (NLS) [Eng75], which offered computer-supported conferencing. Since that time, a variety of CSCW applications have been developed at several research laboratories and universities. *Education* is an inherently cooperative activity involving at least one teacher and pupil.[1] In this paper, we connect the two fields together by taking the reader on a tour of CSCW applications and investigating their use in education.

It can be argued that these two themes are not distinct. In a broad sense, every CSCW application supports education since any group process involves teaching and learning. However, in this paper, we will use education to refer to a learning and teaching process supported by formal educational institutions such as schools and universities. Our discussion will be based more on hypothesis rather than real experience to allow us to investigate the breadth of CSCW applications, few of which have been used together for education or other purposes. See [Sci87] for a specific CSCW application designed and used for education. See [Ell91, Ols90] for previous surveys on CSCW applications and [Bae93] for a collection of papers and in-depth commentaries on CSCW applications.

To make our discussion concrete, we will give several scenarios based on the example of a graduate operating systems course the author has taught. It is a lab-based course and gives students hands-on experience with designing, implementing, and experimenting with the Xinu operating system [Com84]. Teaching assistants manage lab sections and help students with their programming assignments. The assignments are designed collaboratively by the professor and teaching assistants. Students can ask the instructors detailed questions during special office hours and short questions whenever the latter are free. Moreover, they can work in groups on some of these assignments. This collaborative task is currently carried out at a single location. In the remaining discussion, we will investigate how CSCW applications could be used to allow it to be carried out at geographically dispersed sites.

We assume a new hypothetical course taught at two different universities. The course is managed by one professor, located at one of the universities, and two teaching assistants, one at each university. One of them supervises the Tuesday lab section and the other one supervises the Wednesday lab section. Each of these lab sections has students from both universities who use different physical labs located at their universities to perform their assignments.

---

[1] Unless, in the classroom of the future, the human teacher is completely replaced by a computer!

ERIC

The motivation for such a course is straightforward: It allows the students and instructors to work together independent of distance. However, it raises several technical questions regarding the feasibility of such a course. We discuss below how CSCW applications could be used to support such a course. In particular, we discuss how they can be used to support the following processes:

- classroom lecturing
- designing and writing assignments
- lab work
- answering student questions
- surveying and voting.

We show how these applications can be used to facilitate both intra- and inter- site cooperation.

The specific CSCW applications we consider are video walls [Abe90], media space [Har90], the Liveboard [Elr92], the Cognoter [Ste87], the PREP coauthoring system [Neu90], shared window systems [Lau90], the FLECSE collaborative software development environment [Dew93], the GROVE outline editor [Ell91], the TeleConf audio conferencing system [Rie92], shared awareness spaces [Dou92, Man91], the Information Lens [Mal87], the Coordinator [Flo88], computational mail [Bor92], and a voting tool [Dew93]. In our discussion, we assume that each of these applications is cost-effective and works well for the purpose for which it has been designed. This is currently a strong assumption since the area of CSCW applications is still in its infancy.

Class Room Lecturing

The physically dispersed classrooms of our hypothetical course are linked by two-way audio and video connections. Each classroom has a "video wall," [Abe90] which shows the activities in the other classroom, thereby allowing the professor and students in the two class rooms to interact with each other in a media space [Har90]. A camera at each site sends video to a large projection monitor which displays the image. The audio endpoints are speakers and microphones that cover each classroom. Experience at Xerox shows that is is possible to use current technology to establish usable video walls connecting two geographically dispersed sites.

The two classrooms are also equipped with Liveboards [Elr92] connected to each other. Liveboards are large pen-based public computer displays controlled by conventional workstations. They can be connected to each other via a network and used in a shared mode. They currently support several applications including the SlideShow and Whiteboard applications.

The Slideshow application allows the lecturer to show slides stored in the computer. Standing a few feet from the Liveboard, the lecturer gestures with the cordless pen to select the next slide or a random one. The Whiteboard application allows the lecturer and students to use the cordless pen to write down or draw concepts. The two Liveboards are used in the shared mode, thereby allowing the results of the gestures to be seen at both sites.

Designing and Writing Assignments

In the single-site course, the following process is usually followed to design and write assignments:

(1) The lecturer and teaching assistants propose and evaluate various alternatives.

(2) One of the teaching assistants writes up the assignment and gives it to the others.

(3) The others make suggestions which are incorporated by the teaching assistant.

(4) The instructors go through more iterations of steps (2) and (3) if necessary.

In the distributed course, the same process is followed using collaborative applications. The first phase is facilitated using Liveboards and the Cognoter application [Ste87]. The Cognoter allows users to collaboratively brainstorm, organize, and evaluate ideas for the assignments. In the brainstorming phase, the participants propose various ideas for the assignment; in the organizing phase, they collect related ideas into possible assignments; and in the evaluation phase, they evaluate the different assignments. At the end of this activity, they select one of the possible assignments and assign a teaching assistant to write it up.

In step 2, the teaching assistant uses the PREP coauthoring system [Neu90] to write the assingment. PREP is a text editor supporting the abstraction of columns. The teaching assistant uses one of the columns for the assignment and another for justifying the various design decisions he/she made. In step 3, the others use different columns to comment on the assignment and the design decisions.

At the end of this process, the history of the design is saved by these applications, which is used in designing subsequent projects.

Lab Work

Like the classrooms, the labs are equipped with videowalls and networked Liveboards. Like the classroom Liveboards, the lab Videoboards are used by the teaching assistants to explain concepts. In addition, a shared window system [Lau90] executing on the Liveboards is used to demonstrate various aspects of the collaboration-transparent Xinu software and solutions to the programming assignments. A shared window system allows arbitrary, collaboration-transparent, window-based programs to be shared among a set of users by replicating the windows created by these programs on the workstations of these users.

As in the original course, the students use computer workstations to solve their programming assignments. Team members use FLECSE [Dew93] to develop software together. FLECSE is an extension of a conventional software development environment that offers distributed, multiuser software development tools, thereby making it easy to support distributed project teams. For instance, students using FLECSE do not have to huddle together in front of a single workstation to debug assignments— they can instead use a FLECSE multiuser debugger from multiple, possibly distributed, workstations.

The team members also use Liveboards, Cognoter, and PREP to to produce project reports. In addition, they use the GROVE outline editor [Ell91] to collaboratively refine outlines for these reports.

Questions and Answers

As in the original course, students and instructors use electronic mail and the telephone to ask and answer questions. In addition, they use TeleConf [Ric92] to .old audio conferences with the instructors. TeleConf uses the audio capabilities of the participants' workstations to provide computer-controlled conferencing. During office hours of the instructors, the concurrency control method used is moderated floor control. In this mode, the requests of the students are placed in an internal queue visible to the instructor who chooses the next speaker. After a student is finished speaking, the floor always go back to the instructor, who answers the question and then chooses the next speaker.

Often students wish to ask short questions outside office hours. In the original course, they typically went to the labs and instructor offices to see if the instructors were free. In the distributed course, they use a "shared awareness space" [Dou92, Man91] to "walk" to the distributed offices and labs. Images captured by cameras in the offices and labs of participants in this space are switched by server software executing on one of the workstations. Authorized participants can ask the server to bring up images of remote rooms on their wokstations. In our course, the students use this space to see if the instructors are present in their workplaces and free. Conversely, the space can be used by the instructors to monitor the activities of the students!

Electronic mail is still the primary means for asynchronous communication between the students and instructors. In the original course, an answer to a question posed by a student was sent either specifically to that student or to all the students in the class. In this new, presumably bigger, course, Information Lens [Mal87] is used to filter these messages. The tool supports semi-structured mail and supports automatic sorting and categorization of specific and general messages. In our course, it is used to sort messages into exam change notices, classroom change notices, Xinu bug reports, old exams, assignment solutions, requests for class absences, requests for postponement of exams, requests for placing papers/books in the library, and so on. The sorting allows the recipients to prioritize specific messages and find general messages of interest.

It is often difficult for an instructor to keep track of the various conversations with the students and other instructors. In our course, the Coordinator [Flo88] is used to solve this problem. The tool allows, for instance, the composers of requests for placing papers/books in the library to specify the date by which they would want the action completed, and automatically generates reminder messages for the addressee. The Coordinator is not a substitute for regular informal mail— it is used mainly for the more important messages.

Surveying and Voting

A typical course often requires students to fill out survey forms and vote on issues. In our hypothetical course, computational mail [Bor92] and VoteTool [Dew93] are used to automate some of these tasks. Computational mail invokes an associated interactive program on its receipt, which can collect various kinds of information from the recipient. VoteTool allows users to both synchronously and asynchronously vote on various issues. The instructors use computational mail to send messages to the students collecting preferences regarding times of extra classes, labs, and exams. This information is used to propose a set of times and VoteTool is used vote on these times. Without these tools, a large part of class and lab time would be used to resolve these issues.

Conclusions and Future Work

In this paper, we have used the example of a hypothetical distributed course to describe a wide range of CSCW applications and concretely illustrate the benefits of using them in education. In our example, we considered only a two-site course. The importance of computer support for coordination increases as the number of sites increases. Moreover, in our scenario, we maintained the human agents used in the original course. There is considerable interest in investigating how human instructors can be replaced by tutoring software [Rad92, Ade92, Ste92]. Again, the role of computer support for shared software increases when software is used for tutoring students, since such support can allow students to collaboratively learn from the software. Thus the conclusion of this paper is that CSCW has the potential for supporting education by providing mechanisms for supporting classroom lecturing, lab work, assignment design, questions and answers, and student surveys and votes.

Our work is an initial step towards exploring the idea of computer-supported education. The real challenge is to make all of these CSCW applications work together in a cost-effective manner and use them in a real course. This paper provides some of the motivation for undertaking such a project.

Acknowledgments

Many thanks to Prof. Hermann Maurer for suggesting this topic and inviting a paper on it for the conference and to Ronnie Martin for proofreading the paper.

References

[Abe90] M. Abel, "Experiences in Exploratory Distributed Organization," *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*, Lawrence Erlbaum, 1990, pp. 489-510 .

[Ade92] Beth Adelson, "Evocative Agents and Multi-Media Interface Design ," *Proceedings of CHI '92*, 1992, pp. 351-356.

[Bae93] Ron Baecker, *Readings in Groupware and Computer-Supported Cooperative Work*, Morgan Kaufmann, CA, 1993.

[Bor92] Nathaniel S. Borenstein, "Computational Mail as Network Infrastructure for Computer-Supported Cooperative Work," *Proceedings of CSCW '92*, 1992, pp. 67-74.

[Com84] Doug Comer, *Operating System Design - The Xinu Approach*, Prentice-Hall, NJ, 1984.

[Dew93] Prasun Dewan and John Riedl, "Toward Computer-Supported Concurrent Software Engineering," *IEEE Computer* 26:1 (January 1993), pp. 17-27.

[Dou92] Paul Dourish and Sarah Bly, "Portholes: Supporting Awareness in a Distributed Work Group," *Proceedings of CHI '92.*, 1992, pp. 541-547.

[Ell91] Clarence A. Ellis, Simon J. Gibbs, and Gail L. Rein, "Groupware: Some Issues and Experiences," *CACM* 34:1 (January 1991), pp. 38-58.

[Elr92] S. Elrod, R. Bruce, R. Gold, D. Goldberg, F. Halasz, W. Janssen, D. Lee, K. McCall, E. Pederson, K. Pier, J. Tang, and B. Welch, "Liveboard: A Large Interactive Display Supporting Group Meetings, Presentations, and Remote Collaboration," *Proceedings of CHI'92*, 1992, pp. 599-607.

[Eng75] D.C. Engelbart, "NLS Teleconferencing Features," *Proceedings of Fall COMPCON*, September 1975, pp. 173-176.

[Flo88] F. Flores, M. Graves, B. Hartfield, and T. Winograd, "Computer Systems and the Design of Organizational Interaction," *ACM Transactions on Office Information Systems* 6:2 (1988), pp. 153-172.

[Har90] S. Harrison and S. Minneman , "The Media Space: A Research Project into the Use of Video as a Design Medium," *Proceedings of the Conference on Participatory Design*, March 1990, pp. 51-58.

[Lau90] J.C. Lauwers and K.A. Lantz, "Collaboration Awareness in Support of Collaboration Transparency: Requirements for the Next Generation of Shared Window Systems," *Proceedings of ACM CHI'90*, April 1990, pp. 303-312.

[Mal87] T. Malone, K. Grant, F. Furback, S. Brobst, and M. Cohen, "Intelligent Information-Sharing Systems," *CACM* 30:5 (May1987), pp. 390-402.

[Man91] M. Mantei, R. M. Backer, A. J. Sellen, W. A.S. Buxton, T. Milligan, and B. Wellman, "Experiences in the Use of a Media Space," *Proceedings of CHI'91*, April 1991, pp. 203-208.

[Neu90] Christine M. Neuwirth, David S. Kaufer, Ravinder Chandok, and James H. Morris, "Issues in the Design of Computer Support for Co-authoring and Commenting," *Proceedings of ACM Conference on Computer Supported Cooperative Work*, October 1990, pp. 183-195.

[Ols90] Judith S. Olson, Gary M. Olson, Lisbeth A. Mack, and Pierre Wellner, "Concurrent Editing: The Groups Interface," *Proceedings of INTERACT '90*, 1990, pp. 835-840.

[Rad92] Bob Radlinki and Jean McKendree, "Grace Meets the "Real World": Tutoring COBOL as a Second Language," *Proceedings of CHI '92*, 1992, pp. 343-349.

[Rie92] J. Riedl and V. Mashayeki, "Continuous Media in Discrete Objects: Multimedia for Computer Supported Collaborative Work," *Proceedings of 1992 Workshop on Multimedia Information Systems*, February 1992.

[Sci87] J.A. Scigliano, B.A. Centini, and D.L. Joslyn, "A Real-Time Unix-based Electronic Classroom," *Proceedings of the IEEE Southeastcon '87*, 1987.

[Ste87] Mark Stefik, Gregg Foster, Daniel G. Bobrow, Kenneth Kahn, Stan Lanning, and Lucy Suchman, "Beyond the Chalkboard: Computer Support for Collaboration and Problem Solving in Meetings," *CACM* 30:1 (January 1987), pp. 32-47.

[Ste92] Karl E. Steiner and Thomas G. Moher, "Graphic StoryWriter: An Interactive Environment for Emergent Storytelling," *Proceedings of CHI '92*, 1992, pp. 357-363.

Scientific Visualization

GITTA O. DOMIK
Department of Computer Science
University of Colorado
Boulder, CO. 80309-0430, USA

Definitions, Goals and Concepts of Scientific Visualization

The use of visualization in scientific and engineering disciplines is rapidly increasing. Visualization systems offer the tools to augment insights into complex problems, share results with peers, and find their place in the education of various disciplines and at all learning levels. Visualization works with the assumption that transforming information from a non-visual domain (usually quantitative information) to the visual domain stimulates new mental processes that can provide new insights into the interpretation of data/information. Visual thinking enhances our ability to deal with holistic, intuitive and spatial processes, therefore providing a "clearer picture" of complex events and systems.

Definitions and Goals of Scientific Visualization

"Visualization of Scientific Data" describes the application of graphical methods to enhance interpretation and meaning of scientific data, or data of similar characteristics. "Visualization of Scientific Data" will be abbreviated to "Scientific Visualization" or "Visualization" throughout the text of this paper. Data can be derived from various sources, including measuring instruments (e.g. Magnetic Resonance Imaging), as a result of scientific computations performed on supercomputers, digitizing analog information (e.g. maps), or from observations (e.g. algorithm performance).

Data do not become useful until some (or all) of the information they carry is extracted. The goal of scientific visualization is to provide concepts, methods and tools to extract information from data. Visual representations will convey new insights and an improved understanding of physical processes, mathematical concepts and other quantifiable phenomena expressed in the data (Pang, in print). Together with quantitative analysis of data, such as offered by statistical analysis, image and signal processing, visualization attempts to explore ALL information inherent in (scientific) data in the most effective way.

Concepts and tools of scientific visualization are based on other disciplines, such as psychology/perception, human factors, user interfaces and, specifically, computer graphics. While scientific visualization can not be replaced by existing areas of knowledge, it offers more than the sum of knowledge derived from these separate disciplines.

The Process of Mapping Numbers into Pictures

Visualization is essentially a mapping process from one domain (usually numbers) into another domain (pictures) and further into a third domain (the subjective interpretation of the viewer) as pictured in Figure 1.

ERIC

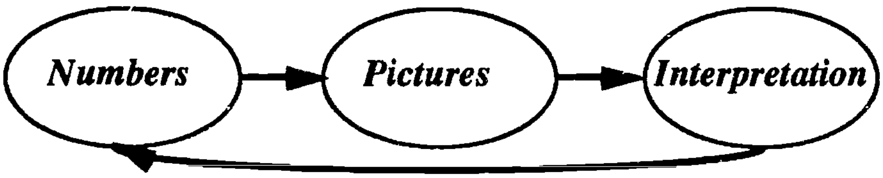


Figure 1. Visualization mapping process

An observer (e.g. scientist) looking at a picture uses this picture as a *vehicle of thinking* (McKim, 1980), but intends to interpret the *meaning* of the *numbers* expressed in the picture. The picture activates mental processes such as the perception of spatial relationships, discovery of patterns or anomalies in large data sets, or the intuitive comprehension of complex processes. These mental processes are obviously different from the ones activated when interpreting numbers without the help of pictures. In a later section of this paper I will address the need of mapping from interpretation (or from the picture) back to the original domain (numbers).

By using visualization the process of data interpretation becomes one step removed from the actual data themselves. If the mapping of "numbers to pictures" is not performed carefully, pictures might not express the true meaning of the underlying numbers, and therefore lead to misinterpretation of scientific facts. Examples of a non-intentional artifact might be a choice of colors that produce abruptly changing hues from continuously increasing numbers. A picture is called *expressive* (Mackinlay, 1986) if it expresses the characteristics of the underlying data values, nothing more and nothing less. To prevent misinterpretations, we therefore strive to create expressive visualizations. Expressiveness is strongly influenced by structure and type of data values.

If the mapping process is not performed purposefully, the resulting pictures might not be *effective* (Mackinlay, 1986) as tools for the interpretation aims the observer has in mind and therefore they will not be useful. The visual representation of a digital elevation model in the form of isolines is very effective when identifying local maxima, but very ineffective when trying to locate south-facing slopes. Effectiveness is strongly influenced by the interpretation goals of the scientist.

While interaction is not the focus in visualization, it is essential to the user in exploring data by the visual mapping process. An in-depth discussion of the need for interaction can be found in (Wilde, 1992).

Visual Cues

Interpretation of pictures or elements of pictures (we use the term "visual cue" to describe elements of pictures) can be subjective. Examples of visual cues are position, animation, color, line angle and slope, length, area, volume. We perceive visual cues as spatial positions and spatial changes, color, density, orientation, size, shape and other perceptual elements. Cognition and interpretation of visual cues depends on culture, education, experience, and individual abilities and disabilities of the viewer. Because a typical picture is the sum of various visual cues, individual differences in interpreting such cues is almost ensured. A typical example for the subjective interpretation of visual cues is color. Because color does not have an inherent quantitative or even ordered meaning (is red "larger" or "higher" than yellow?), different disciplines have forced a different order on color: For astronomers red - yellow - white is synonymous of low - medium - high data values (relating to faint, brighter and very bright stars as seen through a telescope); to a physicist the same order would be true for green - yellow - red (relating to the order of "spectral colors"). A red-green color blind person would find little use with any of the two color schemes.

Some visual cues are more natural for us to interpret than others: If we increase the brightness on a series of objects, we interpret a natural ordering of the information from "low to high," or "less to more." In a similar way a larger histogram bar indicates "more" to the observer. Other visual cues are acquired through education, e.g. the interpretation of street signs, international travel signs, isolines or isosurfaces. Once we learn the meaning of these representations, this knowledge usually stays with us and becomes second nature. The use of natural visual cues is advantageous, because it reduces the danger of misinterpretation. However, natural visual cues are usually too simple to use for the representation of complex information contents. Acquired visual cues are often powerful, yet simple to interpret, once the knowledge to do so has been acquired.

Some visual cues are known to deceive the viewer: e.g. empty space between figures might bring the illusion of shapes (e.g. illusory triangle), black spots might appear in the corners of a white grid on dark background (Hermann grid), lines of same length might not appear as having the same length (e.g. Müller-Lyer illusion), steps of increasing brightness will not look equally increasing (logarithmic eye response and Mach band effect), and the same brightness level or color on the computer screen might appear very different, depending on the background (simultaneous contrast). In all these cases, which can be reviewed in introductory perception books, such as (Sekuler and Blake, 1985), we perceived visual cues different from what is actually presented to the user. Awareness of such pitfalls is necessary to ensure effective visualizations.

In most cases, numerical data to be visualized consists of complex data structures and many parameters. E.g. data for aerodynamic research might consist of pressure (scalar values at 3-d locations), deformation (vectors distributed in 3-d space), and shapes (e.g. airplane). To visualize all available data, various visual cues need to be used. Each individual parameter is mapped onto one or more such cues (e.g. shapes of arrows in 3-d space; color; shaded rendering of airplane): The resulting picture is thus a summary of visual cues, and will, true to Gestalt theory (Gordon, 1989) be perceived as more than the sum of individual cues.

Characterization of Data and Appropriate Visualizations

The type of data values in data sets should influence the appearance of graphical presentations. Data values can be classified as nominal, ordinal and quantitative (Mackinlay, 1986). Nominal values describe members of a certain class, e.g. rocks, wheat, snow; no ordering can be imposed on this class, e.g. "wheat" is not larger or higher or earlier than "snow." Hue and shape are valuable visual cues to express nominal data types. Ordinal values are related to each other by a sense of order, e.g. low, medium, high moisture content. Visual cues used to express ordinal values should depict this order, such as density, brightness, position or size. If color is used, a color bar must be present to assign a certain order onto color. Quantitative values are always expressed as numbers, such as density information available in computer tomography data sets. Even though color is often used to visualize quantitative data sets, this is, in general, a very imprecise visualization of the underlying values. The value in displaying quantitative information via color is that it becomes ordinal information and lets us quickly pick out low, medium or high values.

The underlying data structure of data sets should and will influence the appearance of graphical presentations. Brodlie et al. (1992) propose a classification scheme of data sets that will be used as a basis in this paper to describe dimensions, structures, relationships and dependencies of data. Original definitions have been modified and expanded.

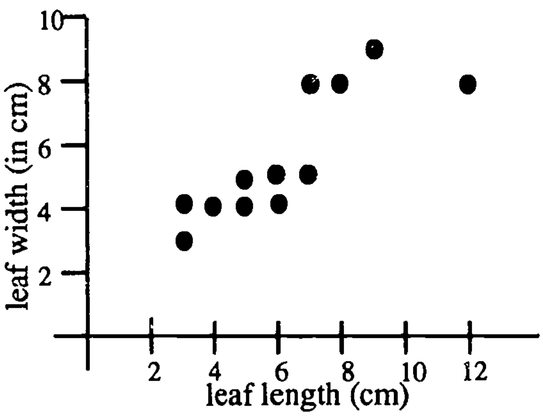
Data sets D are described in the form $D_n^{mC}(d)$, where

- n describes dimensions of a data set;
- C describes the category of data, e.g. S for scalar domains, P for single points, V for vector

data, T for tensor data;

- m describes how many data sets are defined over the same n-dimensional domain or the length of a tuple in a point data set;
- d provides, if necessary, a more detailed description of C. In the case of V or T, d defines the length of each vector, or the size of matrix, respectively. In the case of P or S, "q" (quantitative data type), "o" (ordinal data type), or "n" (nominal data type), or a combination of these data types combined by "+" can be specified.

Simple examples of scientific and engineering data sets are point data sets: D^P describes a list of single points; D^{2P} and D^{3P} are pairs and triplets of numbers, respectively. Scatter plots of one, two or three dimensions (visual cue = position) are typically used to picture this type of information. D^{2P} might be represented as a two-dimensional scatter plot (pairs of numbers define position in two-dimensional coordinate system) or as a one-dimensional scatter plot by assigning symbols to the second number of each point-pair. The choice of graphical representation is dependent on a more detailed description of D: An expressive representation of $D^{2P}(2q)$ pictures a two-dimensional scatter plot (Figure 2), whereas $D^{2P}(q+n)$ is expressed as a one-dimensional scatter plot (Figure 3). The use of glyphs, appropriate to express D^{mP} and D_n^{mS}, has been discussed in (Domik, 1991). Brunswik and Reiter (1938); Chernoff (1973); Pickett and Grinstein (1988); Beddow (1990), and others have documented the effect of complex glyphs on human perception.



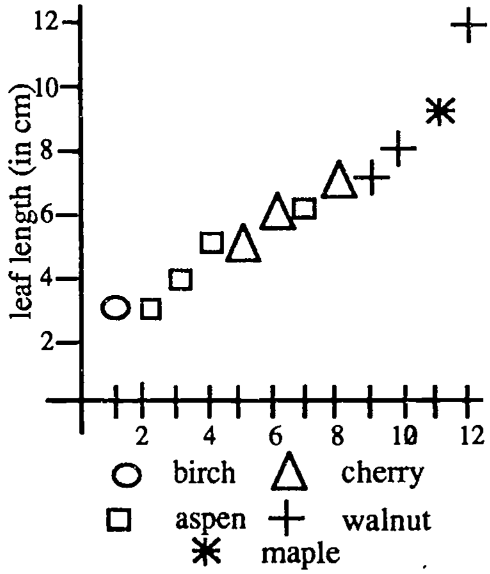


Figure 2 (left): Two-dimensional scatter plot to express $D^{2P}(2q)$

Figure 3 (right): One-dimensional scatter plot with symbols to express $D^{2P}(q+n)$

The above strategy of describing a data set by syntax and semantics before choosing an appropriate visualization applies to scalar, vector and tensor fields in a similar ways as to point data. Data sets of higher dimensions (n, m > 1) may be explored by combining visual cues. Appropriate visualizations of various data sets are listed in (Brodlie et al., 1992) and in (Domik, 1993a).

ERIC

Interpretation Aims and Appropriate Visualizations

Besides a detailed description of data sets to be visualized, the interpretation aim of an observer (e.g. identify local maxima; observe symmetry; locate south-facing slopes) will influence the effectiveness of a visual representation.

While a contour plot will be an effective visual tool to identify areas of a certain threshold value in a data set of type D_2^S, a surface plot of the same data set will be ineffective for the same purpose. In general, a surface plot will be more effective in identifying slopes in a data set than an image display. However, hidden surfaces will prevent viewing part of the data set. Clever use of color tables can enhance image displays for various tasks, e.g. quickly locating values of specific characteristics.

Annotations

A visual representation of a data set might be misinterpreted if certain information is omitted. If color is used, a *color scale* (usually a bar relating color and corresponding data values) must be present. Furthermore, the scaling of world coordinates to screen coordinates must be documented by a *scale bar* if there is any relevance in this information. If geographic data sets are expressed on the screen, *orientation* must be presented to the viewer: e.g. a North arrow, or an indication of the vertical and horizontal directions. The use of animation to express a time series of data must be annotated by a *time indicator*. Too much clutter on the screen will distract from essential information, but omitting explanatory notes and cues might make visual representation useless.

Status-quo of Visualization Systems

Current visualization systems offer computer graphics tools to convert numerical data sets to pictures. They are computer graphics systems rather than visualization systems in that they offer little to enforce expressiveness and effectiveness on resulting pictures. Appropriate combinations of visualization techniques, use of annotations, or avoidance of visual illusions, are the responsibility of the knowledgeable user. Domain experts, e.g. scientists or financial analysts, will fall into many pitfalls as discussed throughout this paper.

Most current visualization tools concentrate on the mapping process from pictures to numbers, and do not address the need of mapping from interpretation (or from the picture) back to the original domain (numbers). In other words, the data values are being lost in the visualization process. But clearly a picture does not carry the precision of information as numbers do, and a combined analysis using visualization and numerical techniques needs to be offered to the observer.

Visualization systems taking into account these user-related issues are currently under research (e.g. Robertson, 1991; Senay, 1992; Kahn et al., 1992).

Educating on Visualization

As scientific visualization brings about new tools to support university education in medicine, the sciences, engineering and other domains, it also needs to be better defined and taught as a discipline of its own. Current education in visualization is strongly dependent on the background of the instructor: Graphics experts teach high-level graphics algorithms; Specialists in perception teach theories of visual perception; Authorities on human-computer-interaction specifically address the graphical component of user interface design. Guidelines on a curriculum in scientific visualization are still in the far distance: Rightly so, because our understanding of opportunities and limitations in this field are still shifting too much to lay out a well defined curriculum. However, we must still provide the educational community with guidelines, even soft ones, to further the goal of arriving at such a curriculum.

Among eleven participants at a recent workshop on education in scientific visualization (Domik, 1993b), the following background of course instructors could be observed: computer graphics, image processing, artificial intelligence, human-computer interaction, psychology/perception, physics, astrophysics. Course audience (undergraduates, graduates, scientists) and course objectives (prepare students to work in Renaissance teams; experiment with effective data exploration; instill an understanding of effective versus non-effective pictorial displays; teach to relate domain-specific tasks to visualization tasks; teach effective communication of scientific results to students, peers, or the scientific community; teach the use and characteristics of output media) were also widely varying. In a survey of current visualization course outlines and suggested visualization textbooks, the following four main topic areas could be found:

(1) Topics classified under category "visualization concepts and human perception" contained these or similar topics:

Definitions and goals of scientific visualization; inspiration and examples; visualization taxonomy; visualization models; history of scientific visualization; increase of information content and data density in pictures; expressiveness and effectiveness; elimination of clutter in pictures; goals, potentials and limitations of visualization; task analysis; "automatic" visualization systems.

Biological, psychophysical, and cognitive aspects of perception; color, spatial and texture perception; visual illusions; theories of perception.

(2) Topics classified as "visualization techniques and computer graphics algorithms" contained these or similar topics:

Interaction techniques; interactive steering; animation; transparency; color spaces; color look-up tables; flow and vector visualization techniques; contouring in two and three dimensions; glyphs; texture mapping.

3-d viewing; perspective and other projections; shading, reflection; hidden surface elimination; ray tracing; surface and volume visualization.

(3) In "computer technology and software systems" hard- and software related issues were discussed. These topics included:

Frame buffer and monitor architectures; neural nets; massively parallel systems; capturing dynamic and static pictures.

Surveys of current visualization software systems, both commercial and public domain software, introduced the current availability of visualization tools.

(4) "Domain specific visualization: Discussion of domain specific data characteristics and visualization techniques" contained these or similar topics:

Data structures, formats, processing, access; data bases and data models; application of visualization to medicine, computational fluid dynamics, multispectral imagery, astrophysics, molecular modeling, climate modeling, process control, network monitoring, neural nets, program visualization and debugging, algorithm visualization, cryptography, 2-d charts.

A statistical approach to this classification scheme shows that topics of type (1) and (2) receive 26% and 37%, topics of type (3) and (4) receive 19% and 18%, respectively, of the overall topics allocation. More than fifty percent of the course outlines emphasized visualization and graphics techniques, with visualization courses relying heavily on computer graphics algorithms.

ERIC

Books showed a more even balance on the topics, due to the fact that computer graphics books were not included in the survey. A summary of course outlines alone (no books) show a distribution of 20%: 43% : 20% : 17% for topics in category (1) through (4), respectively. Visualization course outlines in general lack topics on visualization concepts, reflecting the uncertainty of educators on general definitions and frameworks; topics on human perception are usually taught straight from perception literature with insufficient links to visualization and computer graphics techniques.

A wealth of ongoing research in all four topic areas as defined above and resulting literature in form of conference and journal papers, books and videos is rapidly changing the current status of education in visualization.

The Use of Visualization for Education

Is education in need of changes? Changes in the educational system are prompted by reports of low high school test scores, an increase of facts to be taught to students, a faster turnaround time on knowledge (specifically in technology), and shifting requirements in the workplace.

If there is need for a change in how we teach and learn, does visualization provide (all) the answers for it? The following individual changes seem to be necessary:

- Motivation to learn and acquire knowledge is paramount in younger students. Adults are often motivated by the need for competitiveness in the work force. Young students' decision making is often limited to a choice between studying and playing video games and does not include raising their interaction with the computer to a higher, more creative level.
- A faster learning pace is necessary to learn more in a shorter period of time.
- Work requirements for the future include communication and collaboration skills, together with acquired knowledge (P.J. Denning, 1992).

Interactive visualization provides an active, participatory means to acquiring knowledge, exploring cause and effect of scientific and other phenomena, while adjusting the pace of learning to individual abilities. Passive learning (learning about facts) combines with active experience. Visualization is a channel of communication and a tool for collaboration between teacher and student as well as between peers. Hands-on experience, exploration, communication and collaboration contribute to a better understanding of knowledge and the integration of knowledge into daily life and work. Motivation for younger students while learning should be proportional to their motivation to play video games. Successful examples of the use of visualization in education have included: learning about physical forces acting on objects; the use of Mathematica to explore mathematical and physical phenomena; animation of weather maps to observe complex interactions; exploring topography, populations and economy on a digital, interactive atlas; creating complex pictures from simple programming rules, as in the case of fractals, or exploring math behind well known geometric figures, such as produced by kaleidoscopes; or observing a program's behavior through algorithm animation.

Conclusion

Visualization provides tools aimed at extracting information from non-visual -- often quantitative -- domains. Similar to other areas deriving information from data, such as statistics, the correct use of visualization is important to ensure useful results. While we are expecting the interpretation of visual representations to be intuitive, the encoding process of data to graphical representation is clearly not intuitive. Knowledge of syntax and semantics of data, intents, abilities and limitations of users, as well as the computing environment are important to consider during the encoding process. Education on visualization is therefore essential to its effective and correct use in all areas. While visualization can be helpful on each level of understanding, it is particularly useful as part of the educational system, providing motivation, hands-on experience with science and other knowledge-based areas, and tools for communication and collaboration.

References

Beddow, J., (1990) Shape Coding of Multidimensional Data on a Microcomputer Display, *Proceedings of IEEE Visualization '90*, IEEE Computer Society Press, October pp. 238-237.

Brodlie, K.W., L.A. Carpenter, R.A. Earnshaw, J.R. Gallop, R.J. Hubbold, A.M. Mumford, C.D. Osland, P. Quarendon (Editors), (1992) *Scientific Visualization,Techniques and Applications.* Springer Verlag.

Brunswik, E., and L. Reiter, (1938) Eindrucks-Charaktäre schematischer Gesichter, *Zeitschrift für Psychologie*, 142, pp. 67-134.

Chernoff, H. (1973), The Use of Faces to Represent Points in k-Dimensional Space Graphically, *J. Am. Statistical Assoc.*, Vol. 68, Number 342, 361-368.

Denning, P. J., (1992) Educating a new Engineer, *ACM Communications*, Vol. 35, No. 12, pp. 83-97.

Domik, G. O., (1993a) A Tutorial on Scientific Visualization, *University of Colorado*, Department of Computer Science, Boulder, CO. 80309-0430, USA. In preparation.

Domik, G. O., (1993b) Guidelines for a Curriculum in Scientific Visualization, *Computers and Graphics*, Vol. 17, No. 2.

Domik, G. O., (1991) The Role of Visualization in Understanding Data, *Lecture Notes on Computer Science 555*, Springer Verlag, "New Trends and Results in Computer Science," pp 91-107.

Gordon, I.E., (1989) *Theories of Visual Perception*, John Wiley and Sons.

Kahn, R., J. Bredekamp, Cheeseman, G. Domik, Knighton, L. Treinish, and N. Weir (1992) How Best to Create Interactive Data Analysis Tools That Allow the User to Go from the Pictures to the Numbers, *Special Report to the Communications and Information Systems Division*, NASA.

Mackinlay, J., (1986) Automating the Design of Graphical Presentations of Relational Information, *ACM Trans. on Graphics*, Vol. 5, No. 2, April 1986, pp. 110-141.

McKim, R.H., (1980) *Experiences in Visual Thinking*, PWS Publishers, second edition.

Pang, A., (in print) A Syllabus for Scientific Visualization, to be published in the forthcoming book "Scientific Visualization in Mathematics and Science Teaching", ed. by David A. Thomas (Department of Mathematical Sciences, Montana State University, Bozeman, MT), *Association for the Advancement of Computing in Education*, POB Box 2966, Charlottesville, VA 22902.

Sekuler, R. and R. Blake, (1985) *Perception*, Alfred A. Knopf.

Picket, R.M. and Grinstein, G.G., (1988) Iconographics Displays for Visualizing Multidimensional Data, *Proceedings of the 1988 IEEE Conference on Systems, Man and Cybernetics*, Vol. I, Beijing and Shenyang, People's Republic of China, pp. 514-519.

Robertson, P.K., (1991) A Methodology for Choosing Data Representations, *IEEE Computer Graphics and Applications*, Vol. 11, No. 3, May 1991, pp. 56-68.

Senay, H. and E. Ignatius, (1992) A Knowledge Based System for Scientific Data Visualization, *Technical Report TR-92-79*, CESDIS, Goddard Space Flight Center, Code 930.5, Greenbelt, MD 20771.

Wilde, N., (1992) Truly Interactive Graphics: The Case for a New Cognitive Artifact, *Thesis proposal, University of Colorado*, Department of Computer Science, Boulder, CO. 80309-0430.

ERIC

A Distributed Management System for Multimedia and Hypermedia Educational Resources

Erik Duval, Henk Olivié
Department of Computing Science, Katholieke Universiteit Leuven
Celestijnenlaan 200 A, B-3001 Leuven (Heverlee), Belgium
Email: {Erik.Duval,olivie}@cs.kuleuven.ac.be

1. Introduction

Recent developments have led to considerable interest in multimedia and hypermedia resources *for education.* At the same time, the storage and retrieval of vast amounts of multimedia data (e.g. satellite pictures, medical results, etc.) have become important *database research* topics [DD91, NC91], as traditional database management systems cannot cope well with the new kinds of data involved. A third area influenced deeply by the recent developments is *information retrieval.* The main focus here is on user friendly access.
The ideas and results we will present in the following sections have originated in the projects CAPTIVE, MTS (Multimedia Teleschools) and ACT (Advanced Communication Technologies), funded by the DELTA program of the European Commission. Within these projects, the Computer Science Departments and Audio-Visual Services of University College London and Katholieke Universiteit Leuven have developed a distributed information resource [DOV92, JBD92], consisting of:

- a distributed *server* consisting of:
    1. analogue and digital *multimedia object stores*;
    2. a *central database* about multimedia educational resources (some of these may be stored on the multimedia object stores, but this is not required);
- a *client* that takes care of the interaction with the end user and hides all communication details.
- a *communication infrastructure* for transmission of multimedia data over optical fiber, satellite and/or ISDN and of alphanumerical data over packet switched data networks (PSDN);

In the following section, we will elaborate the fundamental concepts and ideas that underly the development of the information resource.

2. (Re-)Use of Resources

There is an enormous amount of multimedia educational resources available. One of the main problems for someone who might make good use of this material, is the lack of information

ERIC

about what is available, for whom it is intended, what the hard- and software requirements are, where it can be purchased, at what cost, etc.
Moreover, different users have different information needs:

1. A *teacher* or *learner* may be looking for resources to pursue a learning purpose. In this case, the resource granularity (i.e. the relevant architectural unit) will mostly be a complete hyperspace or multimedia course [Nie90]. It might be published on a videodisk or CD-ROM or perhaps users can access it over an electronic network [AMY88].

2. *Authors* of new material, perhaps teachers or learners themselves, can also benefit from what is available.

   (a) The notion of re-use of (substantial parts of) existing resources, so fundamental in the field of software engineering, does not yet seem to have received a lot of attention in the (educational) multimedia and hypermedia community. Adaptation of existing material by the original authors is mentioned in [Nie90], but this is not the same as authoring new resources by adapting and integrating existing material from a different origin into a new context.

   Such an approach may reduce overall costs, as duplication of effort is avoided, and increase quality, as more resources can be invested in the development of the original part of the material. In this case, the resource granularity must be substantially smaller than in the previous case: a few related nodes of a hypermedia network or indeed one single node (perhaps containing a particularly relevant image or graphic) must be described independently as it may have to be extracted from the original context and integrated into a new one.

   (b) Even if re-use is not possible, then an author may still want to have a look at what is available, in order to gain inspiration, or perhaps to learn from the mistakes of others.

In all these cases, users are faced with the difficulty of locating the relevant available resources. It therefore seems appropriate to develop an infrastructure to support management of the available resources. The next section will present an appropriate architecture for such an infrastructure.

3. Architecture of the Resource Management System

The educational multimedia and hypermedia Resource Management System we have developed is composed of three levels.

3.1. Level 1: Multimedia Educational Resources

The resources themselves constitute the first level. They do not need to be stored on digital media: we do not want to exclude analogue videodisks or video tapes, slides and books as they can be very valuable educational resources indeed. A lot of the analogue material may never need to be digitized [Cla91], and, if necessary, transfer on a digital medium is always possible. Digital resources, including CD-ROM, CD-I, CD-Audio, etc. can be more fully integrated into a computerized environment and can be made accessible over electronic networks [AMY88].

Some of the resources will be stored on the digital or analogue multimedia object stores currently under development at University College London. However, as the amount of available resources is enormous, we do not intend to store all the material on these stores.

3.2. Level 2: A Central Database

The second level in our architecture is a central database about the resources at level 1. Its function is comparable to that of a library catalogue, but its scope is not restricted to textual material only. The database includes:

- technical data, e.g. concerning the format and medium;
- persons and organisations involved, e.g. the author or the copyright holder;
- data about the relationship between different resources, e.g. 'is a part of' or 'is hyperlinked with';
- data concerning the content.

An entity-relationship conceptual schema [EN89] describing the data and their interrelationships has been developed. The schema has been translated into a relational schema which has been implemented, using a commercial Relational Database Management System.
We are currently reviewing and elaborating the schema based on the experience gained with the first demonstrator. It is important to note that multimedia and hypermedia objects can be quite complex: different media must remain synchronized, some objects may have links to other objects (typically hypermedia objects, but also e.g. bibliographical references in a text) and some may be isolated, some may be composed of other relevant objects and some may be indivisable, etc.
Moreover, as our aim is to facilitate re-use of educational resources, the *granularity* of description must be quite fine: only a detailed description will make it possible to identify precisely those resources that are appropriate for re-use in a different context.
The relational paradigm for data modeling [EN89], mainly intended for conventional business oriented data, cannot cope with these problems easily. The most important alternative, the object-oriented approach, has a number of drawbacks as well [SSU90]. Therefore we are currently investigating an approach that attempts to integrate some object-oriented modeling concepts into the relational paradigm.

3.3. Level 3: Metadata

The third level holds data about level 2 data, i.e. data about the data describing the resources at level 1. Data about data are commonly called metadata. Although conventional relational database management systems store some metadata in the catalog, much of our work concerns elaborating and extending level 3 data.

1. In older systems, the *data schema*, structuring the data at level 2, is rarely made explicit. It is considered fixed and hard coded into the application programs that access the data at level 2. By making the metadata explicit, application programs can obtain data about the database schema and adapt their retrieval actions accordingly (cfr. section 4.). The result is that when the database schema is modified, the metadata will reflect this and the application programs will not need to be changed.

2. An important issue concerns the domain of an attribute: this is the set of applicable values. Current database systems support only a limited number of predefined data types (integer, real, string, etc.). The domains defined by these data types often include many values that are inapplicable for a particular attribute. If e.g. an attribute Weight is defined as an integer, whereas it is supposed to be a number between 0% and 100% (cfr. infra), then its domain also includes a large number of inappropriate values.

   By extending the type mechanism, we aim at defining domains more accurately. Restricting the domain to those values that are really appropriate will result in a more reliable system:

   (a) when an attempt is made to store inapplicable values (e.g. 'Jacob Nielsen' for the length of a video sequence), this can be detected and refused;
   (b) similarly, when a query is composed, only values belonging to the domain of an attribute are allowed in the search criteria (requesting information about all video sequences with a length smaller than 'Jacob Nielsen' e.g. would be nonsense).

   For each attribute at level 2, the domain definition is stored at level 3. Information about the user defined (abstract) data types is also stored at level 3. Every action involving level 2 data can thus be validated by the Resource Management System.

3. Level 3 also includes information about the *structural relationships* between the values of a domain. The content of level 1 resources e.g. is described at level 2 by relating them to topics. Weights between 0% and 100% can be assigned to indicate the strength of the connection between a resource and a topic. The topics themselves are organised in classifications. The structure of the classification is stored in level 3. It typically corresponds to a hierarchy with more general topics nearer to the root and more specific topics at the leafs.

4. Database Consultation

4.1. Client

The central database component of the server is located at the Katholieke Universiteit Leuven, on a computer accessible over a number of interconnected electronic networks, known as the internet. We have developed a client that supports user friendly access to the database. The client is installed on a local computer that has access to the internet and hides all communication details from the end user. In this way, location transparency is achieved: to the user it appears as if all data are available locally and he is unaware of the exact location(s) where the data are actually stored.
A first demonstrator for the data retrieval tool has been implemented in Hypercard [Goo90]. We are currenly developing a library of C^{++}-functions under Unix that implement more elaborate functionality in a reliable and robust way. We will also develop an X-Windows based user interface to these functions.

4.2. Search Criteria

Search criteria related to content are at the centre of most queries: most teachers, learners or authors are looking for resources about a particular topic. We will first explain why users may find it difficult to express these criteria in a traditional database query or navigation environment. We will then present a solution based on a combination of query and navigation in metadata.

4.2.1. Traditional Query Approach

Databases have mainly been used in highly structured environments, such as business data processing, where queries can be very complicated and may involve massive amounts of data, but where the structure of the data is well known and relatively static. Our environment is totally different: teachers, learners or authors looking for relevant educational resources may have no idea how many and what kind of level 1 materials are available, who created them, when they were created, or the keywords associated with them.
Moreover, it is often problematic to characterize the content of a resource accurately with a limited number of keywords, because

1. their meaning may be unclear without *contextual information*. Resources decribed by the keyword 'food' e.g. may deal with the effect of nutritional habits on life expectancy, or the food production in different countries, or the chemical substances contained in different kind of foods, or appropriate dietetic schemes for specific diseases, or famines in the third world, or national agricultural policies, etc., etc.
2. Users do not know which *keywords* are *actually stored* in the database, which may lead to problems if synonyms are involved. If someone is interested in documents described by the keyword 'manufacturing', then he will probably also want to know wether there are documents in the database described by the keyword 'production'. This leads to problems if there is no adequate support, as the user can never be sure he has included all possible keywords that may describe the documents he is looking for.

4.2.2. Traditional Navigational Approach

Older database technology was based on the hierarchical and network paradigm for data modeling [EN89]. Access to data was carried out navigationally by specifying a path in a query that retrieved one record at a time. Current hypertext and hypermedia systems have combined navigational access with the direct manipulation paradigm for user interfaces: hyperlinks can be followed by activating buttons [Goo90, Nie90].
The problem with this approach is that it is very unflexible: if the database schema is changed, then links that previously existed between data items may disappear and new links may originate. As a result, the path to arrive at a particular piece of information may change completely. Therefore, all application programs that access the data are totally dependent upon the schema: every time it is changed, all application programs must be modified accordingly.
We want to preserve the declarative nature of relational databases: a query should specify exactly *what* the user wants to retrieve. The database management system must take care of *how* the requested data can be retrieved. The following section describes a navigational approach that preserves this declarative nature.

4.2.3. A Combination of Query and Navigation in Metadata

Our approach to data access is based on a combination of query and navigation [DO92]. In order to express search conditions regarding subject matter, navigation in the classifications described at level 3 is supported. Users can identify relevant topics as they navigate in a

ERIC

classification of the content domain. A typical screen during navigation is presented in figure 1.

The small middle box presents the current topic during navigation. The classification in figure 1 is the Medical Subject Headings (MESH), a well elaborated classification for medical topics. The upper box holds a list of more general topics that include the current topic as a subtopic. In a hierarchical structure, the upper box contains only one topic (corresponding to the 'father' node), but in a more general network structure, the box may contain several topics. The big lower box contains a list of subtopics that refine the current topic. Information about the current topic can be obtained with the 'about current' button.

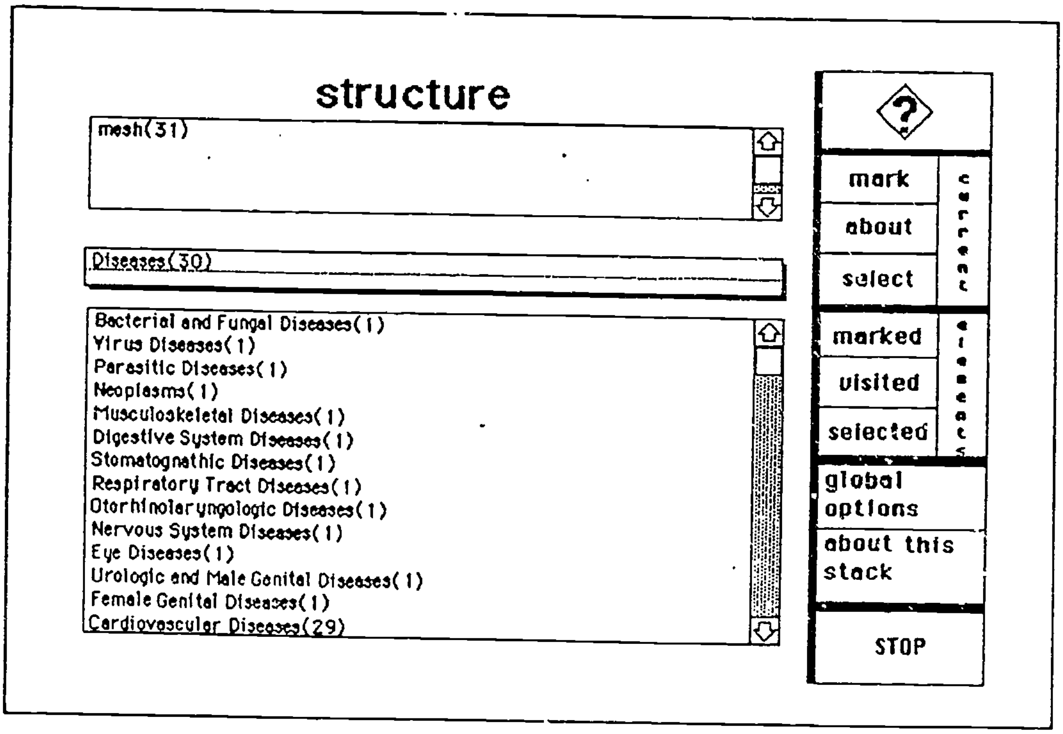


Figure 1: A typical navigation screen

Facilities have been provided to help the user in overcoming the 'Lost in Hyperspace' problem [Nie90]. While navigating, users can mark topics with the 'mark current' button, either permanently (saved when session is ended) or temporarily. If they become disoriented later on, they can retrieve a list of marked topics, using the 'marked elements' button. The 'visited elements' button results in a similar list, containing all topics visited during a session. Both lists function as a lifeline during navigation, enabling the user to go back to one of the topics contained in the list by double clicking on it.

The numbers that appear in round brackets after the topic name indicate how many different learning resources (still image, sequence, video tape, etc.) about the topic are described in the database. This information can be very important as it helps users asses how much relevancy there is, so that they can either refine or broaden their search criteria, before submitting the query to the database, if the number of resources is too high or too low respectively.

A query to the database can be constructed by selecting topics during navigation. The current topic can be included in a list of selected topics with the 'select current' button. Upon activation, the 'selected elements' button displays all selected topics, enabling the user to delete elements from the list, if they wish to do so.

This approach solves the problems encountered when keywords are used in a traditional query approach (cfr. section 4.2.1.:

1. Contextual information is represented by not only displaying the current topic, but also the related ones. The implementation for demonstrative purposes shown in figure 1 does so in a rather rudimentary way: more sophisticated representations are more graphically oriented and display a 'map' of the different topics and their interconnections [Nie90].

2. As users cannot navigate outside the predefined set of topics, they will never leave the domain of appropriate values for the search criteria, whereas this is possible when keywords are used. And, as related topics can be explored, users can make sure that they include all relevant topics.

As users navigate in the *meta*data, the declarative nature of the query approach is preserved (cfr. section 4.2.2.). The search criteria specify *what* must be retrieved from the database. *How* exactly the requested data can be located and retrieved, is the responsiblity of the database management system.

When navigation is finished, additional search constraints can be imposed, regarding technical characteristics, practical considerations (e.g. cost), persons and organisations involved, etc. This is done by filling in forms that present templates of such constraints. The search criteria are then translated into an SQL query by the client. However, this is done transparently: the user will never need to confront SQL. As indicated in figure 1, there is also a help facility (upper right button), and there are some 'global options' concerning the languages used etc., but we will not elaborate on these here.

References

[AMY88] Robert M. Akscyn, Donald L. McCracken, and Elise A Yoder. KMS: A distributed hypermedia system for managing knowledge in organisations. *Communications of the ACM*, 31(7):820–835, July 1988.

[Cla91] David R. Clark. The demise of multimedia. *IEEE Computer Graphics & Applications*, 11(4):75–80, July 1991.

[DD91] M. H. O' Docherty and C. N. Daskalakis. Multimedia information systems – the management and semantic retrieval of all electronic data types. *The Computer Journal*, 34(3), June 1991.

[DO92] Erik Duval and Henk Olivié. 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. Refereed Papers Section.

[DOV92] Erik Duval, Henk Olivié, and Johan Van Heddegem. A database of audio-visual material for education. In Nolan Estes and Michael Thomas, editors, *Education*

"Sans Frontières", Proceedings of the Ninth International Conference on Technology and Education - Paris, 16-20 March 1992, pages 1580–1582, Austin, Texas, 1992. Morgan Printing.

[EN89] Ramez Elmasri and Shamkant B. Navathe. *Fundamentals of Database Systems.* Benjamin/Cummings, 1989.

[Goo90] Danny Goodman. *The complete Hypercard 2.0 handbook.* Bantam books, Toronto, third edition, 1990.

[JBD92] Gordon Jameson, Richard Beckwith, and Erik Duval. Multimedia distribution resource. A research project. *Multimedia Journal on new technologies and learning in Europe*, 6:34–39, May 1992.

[NC91] A. Desai Narasimhalu and Stavros Christodoulakis. Special issue: Multimedia information systems. *IEEE Computer*, 24(10), October 1991.

[Nie90] Jakob Nielsen. *Hypertext & Hypermedia.* Academic Press, 1990.

[SSU90] Avi Silberschatz, Michael Stonebraker, and Jeffrey D. Ullman. Database systems: Achievements and opportunities. *ACM SIGMOD Records*, 19(4):6–22, December 1990. The Lagunita Report of the NSF Invitational Workshop on the Future of Database Systems Research, Palo Alto, California, February 22-23, 1990.

Acknowledgements

If our research has led to any results, then this is largely due to the support of Johan Van Heddegem (K.U.Leuven), Gordon Jameson (U.C.London) and Richard Beckwith (University of London). Without the many lengthy discussions we have had with them, our work would not have been so fruitful, nor would it have been so pleasant.
Many other people have also contributed. Space considerations do not permit us to name them all here, but we would like to mention Nermin Ismail, Piers O' Hanlon and Steve Wilbur at U.C.London and Christoph Liekens and Isabelle Decloedt at K.U.Leuven.

ERIC Full Text Provided by ERIC

Socrates, Aesops and the Computer: Questioning and Storytelling with Multimedia

DANIEL C. EDELSON
Institute for the Learning Sciences, Northwestern University, Evanston, IL 60208, USA

When you think back to the best teachers you've known, you often find that they possess two qualities that make them exceptional. They ask good questions, and they tell good stories. Socrates and Aesops, two of the best known teachers in the western world, are each associated with one of these approaches. The best teachers, however, combine both Socratic and Aesopic[1] teaching in an approach that improves on each of them independently. Taking this observation about effective human teaching to heart, we set out to build a computer-based teaching system that would ask good questions and tell stories in much the same way that good teachers do. To do so requires two elements that are central to both effective question-asking and storytelling: knowledge and timing. If you don't know the right questions to ask or you don't have any helpful stories to tell, you cannot teach effectively with questions or stories. Therefore, teaching with questions and stories requires that you know the right questions to ask and have a sufficiently broad repertoire of stories. Likewise, without timing, knowledge of all the questions and stories in the world is worthless. Students need to be asked questions and told stories at the right moment.

Case-Based Teaching

To combine question-asking and storytelling in a computer-based learning environment, we have drawn upon research in artificial intelligence (AI). The architecture that we use to implement these techniques is called the *case-based teaching* architecture (Schank, 1991). Based in part on the case-based reasoning architecture (Kolodner, Simpson, & Sycara-Cyranski, 1985; Riesbeck & Schank, 1989) from AI, case-based teaching combines the context provided by an engaging task with the power of stories to convey lessons. The case-based teaching architecture is composed of two interdependent components, a task environment and a storyteller. A task environment provides a student with a challenging, naturally motivating task, and an environment in which to explore the task. While a student interacts with the task environment, the storyteller monitors his actions and looks for opportunities to present stories that will help him to learn from his situation. In a case-based teaching system, students learn from their experiences in the task environment and from the stories that they see in response to the situations they find themselves in. A task environment must be carefully engineered to provide students with these opportunities for learning. While the task environment provides students with opportunities for learning, the storyteller must be able to recognize and capitalize upon these opportunities with stories that maximize students' learning. In the system described in this paper, the task environment provides a natural question-and-answer dialogue between computer and student, and the storyteller presents video stories that illustrate the principles that arise in the course of the dialogue.

---

[1] Bill Ferguson of Bolt Beranek & Newman is the originator of the term *Aesopic teaching*.

ERIC

The CreANIMate System

We are exploring the issues of case-based teaching through the development of Creanimate, a learning environment that teaches animal adaptation to 4-7th graders. Creanimate is designed to teach the relationships among the physical features of animals, the actions they perform, and their high-level survival behaviors. The student's task in Creanimate is to design a new animal. Students choose an existing animal to modify in some way, and the task environment leads them through a discussion of important questions relating to the survival of the animal. This task was selected both for its immediate appeal and because it establishes a valuable context for learning. The student becomes invested in the ensuing discussion of survival issues because they apply to his or her own creation. The storyteller in Creanimate draws on a large library of videos of animals in the wild.

Figure 1: A sample screen from Creanimate. The student has just proposed a fish (a pike) with wings. In response, Creanimate initiates a dialogue about the different uses for wings.

A session with Creanimate proceeds in the following way. A student or pair of students begin by selecting an animal and a modification for that animal. They may select these from a list or by typing in their own ideas. For example, one fourth grader asked for a bee with a big nose. In response, Creanimate poses a question about the student's animal, e.g. "How will your bee use its big nose?" Students may either propose answers to that question or ask the computer to suggest answers. The student who asked for a bee with a big nose suggested that it might use its nose to "suck up honey". Answers are typed in free text. In the course of discussing the answers that arise, the storyteller presents videos that show examples of the principles under discussion. For example, the computer could show an elephant using its trunk to suck up water as an example of using a big nose to suck up liquids. When students decide on the answer they prefer, they see a

ERIC

picture of their new animal. After seeing the picture of their animal, the discussion continues with a new question. The new question may be another question about the initial change to the animal or it may follow up on the previous answer. Thus, a subsequent question might be, "How will sucking up honey help the bee to survive?"

The students continue considering issues pertaining to their animal for as long as they wish or until the system is unable to identify an unresolved issue for their animal. Students have a great deal of control over the interaction at all times. Throughout a session, buttons are displayed on the screen that enable students to back up and change an earlier decision, skip forward to a different issue, start a new animal, or ask for help. Figure 1 shows a sample screen from a session with Creanimate.

As this brief summary shows, Socratic-style questioning is the cornerstone of the interaction. In response to a student's proposal, Creanimate asks thought-provoking questions that introduce important considerations. This is accompanied by video storytelling designed to capitalize on the opportunities for learning that arise from the questions posed. Two of the central challenges in implementing systems like these are developing strategies for:

- Identifying appropriate questions to ask.
- Retrieving and presenting appropriate stories at the moment they are relevant.

The solutions we have found to these issues are the subject of this paper, but first I present a brief discussion of the roles of questioning and storytelling in teaching .

Questions in teaching

An important element of expertise is the ability to explain the unfamiliar by asking the right questions. These *explanation questions* (Schank, 1986) are a central focus of Creanimate. The explanation questions for a domain are the questions that are useful for constructing explanations within that domain. Explanation questions are based on the fundamental relationships of a domain. For example, in the animal morphology domain that Creanimate teaches, some explanation questions are, "What actions does the platypus's beak enable it to perform?" and "What combinations of activities are necessary for turtles to reproduce?" Since explanation questions are based on the underlying relationships of a domain, the knowledge of what questions to ask goes hand in hand with understanding the basic principles of a domain. Therefore, one reason for teaching with questions is to model the use of explanation questions to help students learn them. Recent educational research (Collins, Brown & Newman, 1989) has advocated the modeling of desired behaviors in teaching. Both Collins & Stevens (1982) and Brown & Palincsar (1989) have described effective teaching methods—inquiry teaching and reciprocal teaching respectively— in which students developed good questioning techniques after having these behaviors modeled for them.

A second reason for teaching with questions is their value in establishing a context for learning. Posing thought-provoking questions of students (not detail or fill-in-the-blank questions) provides students the opportunity to construct their own hypotheses. A well-designed learning environment can assist students to explore the hypotheses they construct in response to questions and to learn from the successes or failures of their hypotheses. Exploring their own hypotheses provides students with an unequaled opportunity for learning from information they encounter. Students' hypotheses provide them with 1) a frame of reference for evaluating new information, 2) an interest in this new information stemming from their commitment to their own hypotheses, and 3) a framework for integrating new knowledge with their existing knowledge.

To summarize, two reasons for teaching with questions are to help students to learn to ask good questions and to establish meaningful contexts for learning. Creanimate is not the first system to explore questioning in computer-based learning environments. Earlier efforts include Stevens & Collins (1977) and Clancey (1987) who explored questioning strategies on the part of the computer, and Ferguson et al. (1992), Graesser, Langston, and Lang (1991) and Moore & Swartout (1990) who have all developed strategies for allowing students to pose questions.

Stories in teaching

Good parents and good teachers all use stories to convey lessons. Almost everybody remembers learning some important lesson from a fable as a child. Teaching with stories comes as naturally and unconsciously as learning from them. Teaching with stories supports case-based reasoning. The theory of case-based reasoning argues that people solve problems, construct plans, and understand new situations by making reference to similar previous experiences, called cases. In domains as varied as car mechanics, medicine, catering, and architecture, researchers have shown that people solve problems and make decisions by reference to previous cases (Kolodner, 1991). Experience is one way that people enlarge their personal case libraries; hearing stories is another. In this paper, I use *story* to refer to any communication that conveys a case from one individual to another. The term *story* distinguishes cases that are obtained through communication from cases that are obtained through experience. Traditionally, stories have been told orally, but new technologies allow us to convey stories through dramatic visual media.

Since people reason from cases when they deal with complex domains, it follows that good teaching should provide them with sufficiently broad case libraries. A sufficiently broad case library contains a large enough range of cases to allow someone to operate effectively in a domains. While experience is one way to develop a library of cases, it can be inefficient as a means of exposing a student to a broad enough range of cases. As an alternative, cases can be presented secondhand, in the form of stories. To be effective, however, a story must be presented in a meaningful context. Conveying a story in context helps a student to understand its point and to store it in his memory in a way that will allow him to retrieve it at the right time in the future.

Stories can be used two ways in teaching, to create context and to respond to context. The former is the technique employed by the well-known case method of business and law schools, while the latter is what is referred to in this paper as case-based teaching. In the case method, a story is presented to establish a problem situation. Learning takes place through the exploration of solutions to that problem within the context provided by the story. In case-based teaching, the context is provided by a task and stories are presented in the course of exploring that task. Stories that respond to context are presented at the moment that they are relevant to the student's situation and they help the student to learn from his or her situation.

An Architecture for Teaching with Questions and Stories

To ask questions and tell stories effectively, timing is critical. Being asked the right question at the right time or hearing the right story at the right time can cause someone to think more deeply about something, think about it in a new way, or put seemingly disparate elements together. Using insights from research in artificial intelligence, we have developed an architecture that takes advantage of the high information content of video stories and places them in a meaningful learning context through the use of thought-provoking questions. The key to this ability is an underlying knowledge representation that enables a teaching system to identify appropriate questions to ask and present stories that follow through on the learning context established by the questions.

Identifying Appropriate Questions

The following transcript shows two questions that appeared in a Creanimate dialogue:

```
If you could create a new animal by taking an existing animal
and changing it some way, what would you make?
Student:   A FROG THAT CAN FIGHT
That's a good idea, is there a reason why you want your frog to
fight?
Student:  SO IT CAN PROTECT ITS SELF
. . .

How you would like your frog to fight?
Student:   BY JUMPING UP ON HIM
. . .
```

In this dialogue, we see that Creanimate responded to a student's request for a frog that can fight, by identifying two relevant explanation questions. The first question is "Why fight?" and the second is, "How to fight?" Because fighting belongs to a conceptual category that is called a *behavior* in Creanimate, the system identifies two types of relevant questions. The first question is aimed at connecting the behavior *fight*[2] with a survival goal, e.g. *mating*, *eating*, or *avoiding danger*. The second question is aimed at connecting it with a more primitive action, e.g., *biting*, *scratching*, or *running*. Creanimate's knowledge representation enables it to identify appropriate questions. Its representation links together concepts in pairs. Since behaviors in Creanimate's knowledge base can be combined with higher-level survival goals and lower-level actions, the student's request for a frog that can fight triggers two questions aimed at connecting the behavior *fight* with other concepts. (This process is described in detail in Edelson, 1993) After the student resolves the first question, Creanimate poses the second . Subsequent changes lead to new questions. In the transcript above, the student goes on to make his frog fight by biting other animals. This answer in turn raises the question of how it will bite.

By posing appropriate explanation questions in response to students' creations, Creanimate establishes a powerful context for learning. The combination of a student's investment in his animal and the curiosity that is evoked by a thought-provoking question provides an excellent opportunity for teaching with vivid, multimedia stories.

The Right Story at the Right Time

In addition to allowing Creanimate to identify appropriate questions to ask, the expressiveness of the knowledge representation in Creanimate enables it to evaluate students' open-ended responses to questions and to present appropriate video stories in response. Consider the following fragment taken from the same transcript as the previous example:

```
That's a good idea, is there a reason why you want your frog to
fight?
```

---

[2] In this paper, concepts from the Creanimate knowledge base appear in italics.

ERIC

```
Student: SO IT CAN PROTECT ITS SELF
That's a good idea. Bees protect themselves by fighting enemies.
Would you like to see that?
Student: YES
[VIDEO: "BEES EXPEL WASP"]
```

In this example, the system examines its knowledge base and ascertains that *protecting itself* is one of the behaviors supported by *fighting*. In addition, it recognizes that it has a video that shows bees fighting enemies to protect themselves, and it offers the student the opportunity to see this story confirming his hypothesis. The Creanimate storyteller retrieves this story by searching the indices in its story library. An index in a case-based teaching system is a label for a story that contains information about the circumstances in which the story would be appropriate to tell a student.

In the Creanimate system, an important part of the index for every story describes the content of that video. In the case of the *Bees Expel Wasp* story in the example above, the index tells the system that in this video, the animal *bee* is shown *fighting enemies* ***in order*** *to protect itself*. This index contains a great deal of other information including the fact that these bees use their *stingers* ***in order*** *to sting* and that they *work together* ***in order*** *to fight enemies*. However, this additional information is not relevant to the current explanation question and the student's current answer. An important aspect of the Creanimate knowledge representation is its information is stored in abstraction hierarchies. Therefore, the system knows that *fighting enemies*, *fighting predators*, and *fighting males for dominance* are all types of *fighting*. So, when a student asks for an animal that fights, it is able to suggest or recognize each of these as reasons for the student's animal to fight. When the student answered the question, "Why fight?" with "protect itself", Creanimate was able to ascertain both that protecting oneself is a reason for fighting, and that the *Bees Expel Wasp* story shows a specific example of that.

Of course, students respond in different ways to Creanimate's questions. Some students are timid in proposing answers or may not have any ideas, so Creanimate gives them the opportunity to say, "Give me some suggestions." When they do, Creanimate shows stories that suggest possible answers. Students can see several answers to each explanation question and pick the one that they like for their animal. Alternatively, students sometimes propose answers that Creanimate can not verify as correct. These unverifiable answers are also treated as opportunities to show stories, as in the following transcript.

```
How would you like your frog to fight?
Student: BY JUMPING UP ON HIM
I don't know any animals that jump to fight, but I do know why
some animals jump.
Salmon jump. Do you know why salmon jump?
(I think this video is funny. )
```

The process by which Creanimate identifies relevant stories is called *reminding*, and the system includes several reminding strategies. The two that were shown above are called *example reminding* strategies because they present examples of the answers being discussed. In addition to example remindings, Creanimate also employs *incidental remindings*. Incidental remindings are peripheral to the main discussion of the explanation question, but they help expose students to new things in the same way that human teachers do. The following is an example of an incidental reminding from a different Creanimate dialogue:

```
...What would you make?
Student:    A bear with wings
Actually there are mammals that have wings.  For example, flying
foxes have leathery wings.
Is that something you would like to see?
```

This is an example of an incidental reminding strategy called *expectation-violation* reminding, because the story violates the expectation that mammals do not have wings. To allow Creanimate to present remindings of this sort, indices contain information about expectations that are violated by a particular story. The reminding strategies in Creanimate are described in more detail in Edelson (1992;1993).

Results and Conclusion

The system described in this paper has been implemented as a prototype and is currently undergoing testing. The transcripts presented in this paper were all drawn from actual student sessions. Evaluations of the system are being conducted to inform the next stages of development and to evaluate the motivational and instructional effectiveness of the system. To date, Creanimate has been tested in two controlled evaluations involving a total of approximately 60 students and has been used by hundreds of individuals in informal settings. The response to the system by both students and teachers has been extremely favorable. Students maintain complete concentration for sessions of 45 minutes or longer. Their reactions to the system are dramatic. They react to the stories with a range of emotions including delight, fear, revulsion, and amazement. These reactions illustrate the power of video stories to captivate children and enhance their learning.

What transcripts and descriptions fail to convey, of course, is the drama of the stories that Creanimate presents. The videos of animals in the wild employed by Creanimate are an excellent example of the superiority of visual media over less dynamic techniques for conveying knowledge. Children and adults alike find these clips riveting. The challenge of constructing multimedia systems like Creanimate is placing this powerful media in a meaningful context for learning. In Creanimate, the explanation questions concerning students' animals establish this context, and the reminding strategies present stories that capitalize on this context. With Creanimate, students do not receive information in a predetermined, inflexible order as in traditional, linear video, and they do not see it in a detached context in which they have no motivation for learning. Rather, they see video clips in the context of a discussion in which they are invested, in response to questions that encourage them to construct and explore their own hypotheses. Students' discussions provide them with both a perspective from which to integrate new knowledge and a motivation to absorb it. The vividness of the media assists them to retain it. The aim of Creanimate is to provide students with both an understanding of the basic relationships that underlie animal adaptation and cases to reason with when they think about this subject. The questions teach relationships and the stories provide cases.

References

Brown, A.L. & Palincsar, A.S. (1989). Guided, Cooperative Learning and Individual Knowledge Acquisition. In Resnick, L. B. (Ed.) *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*. Hillsdale, NJ: Lawrence Erlbaum Associates. 393-451.

ERIC

Clancey, W.J. (1987). *Knowledge-Based Tutoring: The GUIDON Program.* Cambridge, MA: MIT Press.

Collins, A. & Stevens, A. L. (1982). Goals and Strategies of Inquiry Teachers. In R. Glaser (Ed.), *Advances in Instructional Psychology II.* Hillsdale, NJ: Erlbaum.

Collins, A., Brown, J.S., and Newman, D. 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.* Hillsdale, NJ: Lawrence Erlbaum Associates. 453-494.

Edelson, D. C. (1992). When Should a Cheetah Remind You of a Bat? Reminding in Case-based Teaching. *Proceedings of the Tenth National Conference on Artificial Intelligence.* Cambridge, MA: MIT Press.

Edelson, D. C. (1993). Learning from Stories: Indexing and Reminding in a Socratic Case-Based Teaching System for Elementary School Biology. Ph.D. Dissertation, Northwestern University.

Ferguson, W., Bareiss, R., Birnbaum, L., Osgood, R. ASK Systems: An Approach to the Realization of Story-based Teachers. Institute for the Learning Sciences Technical Report no. 22, April, 1992, Northwestern University.

Graesser, A.C., Langston, M. C., and Lang, K. L. (1991) Designing Educational Software Around Questioning. *The International Conference on the Learning Sciences: Proceedings of the 1991 Conference.* Charlottesville, VA: Association for the Advancement of Computing in Education. 204-210.

Kolodner, J. L. (1991). Improving Human Decision Making through Case-based Decision Aiding. *AI Magazine* **12** (2). 52-67.

Kolodner, J.L., Simpson, R.L., and Sycara-Cyranski, K. (1985). A Process Model of Case-Based Reasoning in Problem-Solving. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence.*

Moore, J. D. & Swartout, W.R. (1990) Pointing: A Way Toward Explanation Dialogue. *AAAI-90 Proceedings Eighth National Conference on Artificial Intelligence.* Cambridge, MA: MIT Press. 457-464.

Riesbeck, C. K. & Schank, R. C. (1989). *Inside Case-Based Reasoning.* Hillsdale, NJ: Lawrence Earlbaum.

Schank, R. C. (1986). *Explanation Patterns: Understanding Mechanically and Creatively.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Schank, R.C. (1991). Case-Based Teaching: Four Experiences in Educational Software Design. Institute for the Learning Sciences Technical Report #7, Northwestern University.

Stevens, A.L. & Collins, A. (1977). The Goal Structure of a Socratic Tutor. *Proceedings of the National ACM Conference, Seattle Washington.* New York: Association for Computing Machinery. 256-263.

Acknowledgements

This work is supported in part by a grant from the IBM Program for Innovative Uses of Information Technology in K-12 education, and by the Defense Advanced Research Projects Agency monitored by the Office of Naval Research under contract N00014-91-J4092. The Institute for the Learning Sciences was established in 1989 with the support of Andersen Consulting.

The CreANIMate team includes, in addition to the author, Bob Kaeding, Ken Greenlee, Riad Mohammed, Diane Schwartz, John Cleave, and Will Fitzgerald.

COMPUTER BASED EDUCATION AT QUEENSLAND UNIVERSITY OF TECHNOLOGY: WHY MOVE TOWARDS MULTI-MEDIA?

H. DAN ELLIS and CHRIS OWEN
Computer Based Education Section
Queensland University of Technology
GPO Box 2434, Brisbane, QLD 4001
Australia

Background

In 1984, the author applied for six months secondment from lecturing in physics to spend some time looking at how computers could be used in physics education at university level. In the beginning, the motivation was certainly that of wanting to spend a peaceful time away from the hurley-burley of the teaching year, playing with computers. That was not to be, for it was put to me by my Head of Department, quite forcefully, that I might achieve something useful by seeking out success stories from physics departments in other universities to do with their own uses of computers in support of teaching and learning. Although this sounded a most unexciting and uninteresting task, I succumbed to pressure and sought help from the reference librarians to track down university physics departments with more than a handful of physics graduates per year, and where English was used in communication. The six hundred departments fitting these criteria were then sent a questionnaire.

Of the one hundred and fifty replies, the most striking and useful was that from the United States Air Force Academy, in Colorado Springs. With a very large cadet intake numbering some 1,200, they had experienced for years rather poor examination results in first year physics. They achieved very marked improvements in examination performance by giving their students access to numeric variable physics problems, delivered via a simple network to the screens of a number of IBM PCs. Improvements of 60% to 100% in examination performance were reported in parts of the curriculum where the computer exercises were available, with no change in exam performance where no computer tutorials had been developed.[1]

This remarkable story of success formed the basis of arguments which I then began to put forward to senior management within QUT. In essence, these arguments were:

- with rapidly increasing enrolments, and funding per student decreasing, class sizes were increasing, and teaching staff were having increasing difficulty coping with the challenges presented by large classes;
- evidence was now available to show that low-cost, off-the-shelf, industry standard computer technology could be used in large classes to produce marked improvements in examination marks in very large first year university physics classes;
- as a technological institution, the community at large had an expectation that we would be involved in using computers in the education process;
- if we were to be well advanced in CBE in, say, ten years time, we needed to make a start soon.

In mid-1985, following a wide range of consultations and discussion papers, funding was found for a pilot project in CBE. Six months later, following an intensive period of laying down the basics of how the project would be set up, managed etc., investigating networking solutions, authoring systems and other sundry items, twenty computers were delivered at 3 pm on Christmas Eve, 1985.

The Beginnings

By early February 1986, the computers were installed in a room in the Library, were connected to a Novell server, and first year undergraduate engineering students were using numeric variable question-and-answer tutorials, adapted from pre-existing paper-based tutorial exercises. The XT-compatible computers had CGA color screens and booted from a floppy disk given to the student in exchange for his or her student identity card from a Library loans desk adjacent to the CBE Facility, as it was called. Each tutorial had a number of questions, each question had a help screen, and students' scores were recorded. Lecturers were provided with print-outs of students' scores each week. Tutorials were available only for a limited period, to encourage students to work steadily throughout semester. Students could book a machine at a time of their own choosing, and could obtain a print-out of questions they had failed to answer correctly, together with their answer and the correct answer. Students were always given the correct answer on screen. Substantial improvements in examination performance (from 47 to 56%) were obtained at the end of first semester, compared with previous years.

Within a few weeks, lecturers from other disciplines had begun to ask if similar projects could be set up for them - question-and-answer tutorials aimed at revision, drill-and-practice, and self-assessment for the students. All wanted weekly reports of student progress, which could be automatically generated. By the end of 1986, thirteen projects were active or in development, and 10,000 hours of screen-based tutorials had been delivered. Clearly, some hidden seam of demand had been tapped.

Since 1986, the QUT CBE project has grown enormously due to a number of factors, including strong support from the management of QUT, the choice of good networking software (Novell Netware), realising the importance of systems support staff to look after delivery issues, and choosing an authoring system with excellent customer support (Microcraft's AUTHOR). Growth of usage can be shown in the following table:

	1986	1990	1992
Staff (F/T equiv.)	1½	10	22
PCs delivering CBE	20	60	400+
Student Usage (Hours)	10,000	98,000	219,000
File-server	80186 (1x40Mb)	80386 (400Mb)	3x80486 (1Gb)

In 1993, CBE Section is a service organisation comparable with the university's Library or Computing Services organisations. CBE services have been extended to multiple labs on five campuses at QUT; this number is growing all the time and in the future any PC attached to the campus backbones will have access to CBE fileservers.

Production Methods

The approach to the production of software has undergone substantial change over the past two years. Development and consulting work for outside organisations has precipitated the development of new standards for production of CBE materials for the Section. Graphic artists have been employed to concentrate on high quality screen design. The standards developed contain a number of on-screen features which enhance the "look and feel" of the materials. The graphic artists join applications programmers, operations and systems staff, quality reviewers and instructional designers to form development teams, including lecturers as content and pedagogy experts. The graphic artists have produced a Style Manual which explains how graphic design principles can be applied to screen-based CBE development. This manual includes explanations of preferred colours, colour combinations, and layouts for computer screens. Previous papers have detailed other aspects of the QUT approach (see bibliography).

Instructionally, QUT's CBE projects have traditionally been question and answer tutorials which allow students in large classes to do reinforcement, self-assessment and revision. Reports are sent to lecturers weekly giving information on students' efforts, and on questions that consistently prove difficult. Some projects have more instruction built into them in the form of reference and help screens. On-screen instruction has proved to be more expensive for Faculties than question-and-answer materials, since CBE Section makes a cost recovery charge on all development work. Given growing student numbers and limited resources for the Faculties, the cost of screen-based instruction may prove a short-lived barrier to its use, since programmer productivity is rapidly improving with standardised procedures, higher level and faster authoring systems and the decreasing costs of technology.

Authoring tools in use other than AUTHOR are Multimedia ToolBook, IconAuthor, and Questionmark by BBC Software. Evaluations of others are currently taking place. Projects being developed in ToolBook include nursing case study approaches using decision paths and situations where students can write text to a data base for marking by a lecturer. Questionmark has been used for multiple choice examinations where movement through the questions is flexible. IconAuthor is the basis of our video-disc based language learning labs.

At the present time the Section is becoming involved in the use of video and sound in CBE tutorials as well as CD-ROM as an access alternative. Two networked multimedia labs, based on interactive videodisk and CD-ROM drives with Soundblaster cards are under construction at the time of writing, and progress will be discussed at the conference. The primary use of these labs will initially be for foreign language teaching, based on US Air Force Academy methodologies. However, QUT's pedagogical aims are different from those of the Academy, and at the time of presentation of this paper, newly developed software aimed at giving lecturers simple and time-efficient methods for software production should be available for demonstration.

Multi-Media: The Pros and Cons

Pressure from a number of sources including vendors and teaching staff are pushing QUT towards exploiting multi-media in CBE. Admittedly, multi-media can do a number of things that are impossible or very difficult to do by any other means; Stan Smith's classic videodisc from the University of Illinois allowing students to mix dangerous chemicals springs to mind. Simulations, processes, guided exploration of inherently multi-media data such as sound or film archives, and cultural aspects of teaching, such as are necessary in foreign language learning, are all excellent ways of delivering quality educational and training experiences. In education, the down side is - COST, including the additional development time needed.

Material for use by trainee aircraft pilots is cost-effective even if it costs, say, $100,000 per hour of material, and is delivered on a $3,000,000 simulator. With first year business students learning the principles of accountancy the cost-effectiveness equation is very different. For one thing, there are thousands of students, requiring hundreds of delivery machines. For another thing, in some areas, the content of the material is quite ephemeral. My university will not give me large sums of money to spend on developing expensive materials to educate a few students in an area where the content will change next year, meaning substantial modification. What it DOES give me is large sums of money to develop lots of simple material (capable of being

ERIC

delivered and managed on low-cost PC networks) for thousands of students in content-stable areas. And there's the rub. The education budget is smaller than the military budget; it's harder to justify multi-media. We will, however, be moving further into multi-media, in specific projects where the cost can be justified. We will certainly be producing our own CD-ROMs with digitised images and audio and data, for sale in the university's bookshops.

Delivery System

The delivery system used for CBE at QUT has evolved over a number of years to meet the following needs:

- Good response time. Students who have to wait for screen updates lose concentration and interest.
- Minimal down time. CBE material is available from a number of labs. A crashed file server or faulty cabling can cause loss of assessable results for students working on the system at the time of the fault, and cause frustration and hinder acceptance for others who try to use the system when it is not available.
- Transparent access from any lab or campus. Simple and uniform access allows use from an unfamiliar lab when other labs are busy. Students can access the material from the campus which is closest to them. In many cases this is not the same campus at which they attend lectures.
- Integrity of collected results. Academic staff often use the results obtained via CBE for student assessment, or as a pointer to students with problems. We need to minimise the opportunity for students to hack into and corrupt or fake results.
- Virus prevention. Shared personal computing facilities can often act as computer virus incubators. Careful use of disk and network access rights can go a long way to reducing the risks.
- Ease of use. Users come from a wide range of backgrounds; many seem never to have seen a computer before. The system needs to be very simple to use, and there needs to be adequate help available to assist the computer-phobic user.
- Ease of management. Higher education, more so than other industries, needs to make better use of technology to improve productivity. A high priority has been placed on automating as many processes as possible, and making things easy to manage to reduce support staffing needs.

The current system

Each set of materials constitutes a *project* with an individual access account on a Novell file server. This account has the minimum network access rights needed to run the software.

Access is only available via a *Project Menu* access program. This calculates and displays an appropriate menu, obtains the student number, and passes it to the project software. It also controls concurrent access restrictions for programs which have a limited licensing arrangement, or which have a low priority. Access via other means is prevented by having all accounts protected by a randomly encrypted password. The Project Menu is the only program which can decrypt these passwords and log the user in. This ensures that setup software is run, and helps prevent the user having direct access to writable files from DOS.

PCs in labs owned and operated by external departments get local laboratory specific services from a file server managed by the same department. This file server may be Novell NetWare, Digital Pathworks or Microsoft LAN Manager based. In the case of Novell NetWare based labs, we call this file server the *home* file server. For these labs, a standard script is executed from the *home* file server. This sets up local services such as printing, temporary writable work space, and the location of the command shell. This arrangement allows laboratory specific services to be managed independently by the faculties operating the laboratories.

In the case of Pathworks or LAN Manager based labs, the Novell protocol stack is also loaded on these PCs. Digital's DLL or Microsoft's NDIS layer is used to do the protocol multiplexing necessary to allow sharing of the Ethernet card by the two protocol stacks. For these labs, the local services for the command shell, printing and workspace must be set up before logging into the CBE file server.

There is one CBE file server for each campus. Using NetWare Name Service, these are managed as a single domain. This means that all created accounts appear on each file server, and have the same login scripts and passwords. A program running every few hours synchronises courseware files and access rights across the servers automatically.

Each server services a local laboratory via a direct Ethernet connection, and a number of other labs via the campus Ethernet network. A campus specific batch file locally stored for each lab, CBE.BAT, tries to attach to the closest CBE server. If that is not available, one from another campus will be used transparently. All PCs in public labs are equipped with boot PROMs, which helps to reduce the incidence of computer virus infections.

Backups are performed to Video 8 tape installed on one of the servers. A full backup of this server is performed, with only the smaller server specific areas of the other servers being backed up across the inter-campus network link. This results in minimal network traffic due to backups.

A crude e-mail facility is available for students to report problems and give feedback to operations staff. The ability for staff to send messages to individual users is also available. Users cannot send messages to each other.

An automated booking system runs on the lab operated by CBE Section. This lab contains 80 PCs, and is used very heavily throughout semester. The booking system smooths demand and reduces the need for queuing. Students can book PCs for individual use up to two weeks in advance. Staff can book a group of PCs for class use. Modifications to the boot PROMs prevent users from booting via floppy disk and bypassing the booking system's authorisation, and also prevent the spread of boot sector viruses. Disk access can be enabled later via software if needed. A small resident timer program set by the booking system locks up the PC at the end of the booking after providing a 10 minute warning. This forces the students to be re-authorised by the booking system and gives others who have pre-booked the chance to use the system. A bookings-only PC is available for making bookings when the lab is full.

Student enrolment details are obtained from the University's Student Information System automatically each evening. A list of required subjects is maintained on the CBE file server. A process running under UNIX on the administration system obtains these requests via FTP, extracts the student details required from the master database, and transfers the results back via FTP. The information is imported into Author's enrolment database each day, so our details are never much more than a day behind those of the central University system. At the start of semester when central enrolment processing is still occurring, some details are put in by hand as required.

Second or third year Information Technology students are employed casually as Lab Advisors to assist users with various problems from finding the power switch to diagnosing configuration problems and recovering lost or corrupted files. This helps users feel more comfortable, especially at the beginning of the semester when there are many new users.

An audit trail of usage is logged by the file servers. Reports can be obtained showing usage by campus, lab, project, or time. This is used in the management of the labs, network and file servers for predicting growth.

Reports on student results from CBE exercises are produced regularly and mailed to academic staff involved with the subject.

DEET Study on the Impact of CBE on QUT

In 1991 the Commonwealth Government funded a study of the use of Computer Based Education in Australian Higher Education with particular reference to the QUT CBE project through the Evaluations and Investigations Program of the Department of Employment, Education and Training. The report from this year long study will shortly be published by the Australian Government Publishing Service. The study includes surveys of Australian higher education institutions, of QUT staff and QUT students.

The following extracts, published with the permission of the Commonwealth of Australia, provide a broad picture of the report's findings.

"Student users of computer based education at QUT use four kinds of material; question and answer tutorials as part of a subject, other learning materials such as simulations, commercial software packages such as WordPerfect, and information services such as databases on CD-ROM. The student survey examined the use of each of these four, finding that the great majority of students use question and answer tutorials and that this number is significantly more than those using the other services.

Data provided by students on the extent to which their use of CBE helped to keep abreast of work, was used to follow up lectures, was used for revision, etc are provided. Overall the student assessment, which is qualitative as well as quantitative, is positive to an unanticipated degree, with the main dissatisfactions being expressed with the accessibility of the service rather than the nature of the learning experience provided by it."

"The nature of involvement by teaching staff is described, including the principal motivation for the adoption of CBE as a teaching technique. Staff indicate a range of responses on issues such as the amount of teaching time involved in the use of CBE, the amount of time spent marking, the extent to which CBE has changed teaching workloads, the extent that CBE has changed interaction with students, affected student learning, affected student performance etc. The assessment of staff attitudes to the use of CBE is strongly positive, a similar finding to that concerning student attitudes."

Summary

Six years down the track, CBE at QUT is no longer an experiment. It is a heavily used service which is as much a part of the educational fabric of the university as lectures or the Library. In its future is increasing uses of multi-media, aimed very pragmatically at areas where it pays! As the executive summary of the DEET report previously mentioned puts it:

"Computer based education has been the subject of debate for some time. Its implementation at QUT in terms of delivery, and high degree of student exposure is somewhat unusual. It has attracted considerable interest from other higher education institutions both within Australia and overseas. In the context of developments elsewhere, the case of QUT is distinguished to some degree by its concentration on service and delivery to a proportionately greater degree than on production and invention. ....the rapid growth of CBE at this institution of higher education may, in time, come to be seen as a precursor to much more pervasive paradigm shifts in education towards greater learner centred activity and educational outcomes of ever improving quality."

Bibliography:

Ellis, H.D. "Technology in Education: White Elephant or Great White Hope?" Keynote address, ITTE '92 (Information Technology for Training and Education), 29 September-2 October, 1992. In Conference Proceedings, published by University of Queensland Press, ed. A. Holzl and D. Robb. pp. 31-42.

ERIC

Ellis, H.D. (1992). CBI Research and Development Centers: Queensland University of Technology. *Journal of Computer-Based Instruction*, Season 1991, 18, (4), 139.
Ellis, H.D. (1992). Low-cost, High-volume Exploitation of Computer Technology in Education. *Journal of Computer-Based Instruction*, Summer 1992, 19, (3), 73-76.
Ellis, H.D. "Computer Based Education at Queensland Institute of Technology." Proceedings of the Fourth Annual Computer Assisted Learning in Tertiary Education Conference, Adelaide 1986, pp 92-96.
Ellis, H.D. "Computer Based Education at QIT - A Progress Report." ASCILITE-88: COMPUTERS IN LEARNING IN TERTIARY EDUCATION: Proceedings of the Sixth Annual Conference, Canberra 1988, pp 342-349. Ed. Fielden, Hicks, and Scott.
Ellis, H.D. and Wong, C.F. "Relativity Tutorial - A Use of Simulation." ibid., pp. 137-140.
Kokot, S. and Ellis, H.D. "A Useful Working CBE Chemistry Program based on the Microcraft AUTHOR System.". ibid., pp 175-179.
Ellis, H.D. "Some Recent Major Developments in Computer Based Education at Queensland University of Technology." ASCILITE-89: Proceedings of the Seventh Annual Conference of the Australian Society for Computers in Learning in Tertiary Education, Bond University 1989, pp 186-195. Ed. Bishop and Baker.
Ellis, H.D. "Using Computers in University Education - Experiences from Computer Based Education at Queensland University of Technology." CATS'90: Proceedings of the International Conference on Computer Aided Training in Science and Technology, 9-13 July 1990, Barcelona, Spain, pp 170-175. Ed. E. Onate et al.
Ellis, H.D. "The QUT CBE Project: Exploiting Technology for Productivity in Education." ASCILITE-91: Proceedings of the 8th Annual Conference of the Australian Society for Computers in Learning in Tertiary Education, Launceston, December 8-13, 1991, pp 167-173. Ed. R. Godfrey.
Ellis, H.D. "Using Computer Technology in Higher Education: A World Tour." ibid., pp 175-182.
Winn, J. and Ellis, H.D. "The QUT CBE Project: A Brief Analysis." ibid., pp 705-713.
Cochrane, T., Ellis, H.D. and Johnston, S.J.. "Computer Based Education in Australian Higher Education - a Case Study at the Queensland University of Technology." Australian Government Publishing Service, Canberra. (Accepted for publication April 1992).

[1] US Air Force Academy, 1985. Private communication.

ERIC

Trends in Computer Graphics

DIETER W. FELLNER
Department of Computer Science,
University of Bonn, D-5300 Bonn 1, Germany

Advances in the development of image synthesis algorithms, hardware processing power, user interfaces, multithreaded operating systems and integrated media have introduced graphics techniques to almost every discipline as well as fueled demand for graphics applications that take advantage of these capabilities. In fact, 3D graphics will most likely become a mainstream desktop (workstation as well as PC) technology. In spite of the valuable progress that has been achieved in the field of computer graphics, a number of important problems remains to be addressed. Focusing on image synthesis (i.e. excluding multimedia and visualization) we will discuss some recent developments.

Rendering

Image synthesis has gradually evolved from heuristic shading models (flat, Gouraud, Phong shading) to models with a basis in physical reality. By the first half of the last decade physically based reflection models had been applied to computer rendering [3, 10]. Distributed ray tracing provided a common framework within which a large number of previously difficult physical effects could be simulated, including gloss, penumbra, depth of field, and motion blur [9]. In spite of many successes, however, image synthesis remained unable to simulate indirect 'ambient' illumination provided by light reflected among the many diffuse surfaces that typically make up an environment.

One important consequence of the trend towards physical accuracy was an increase in the attention paid to results available in the fields of physics and engineering. It was thermal engineering that finally provided the basis for radiosity, the first image synthesis approach to accurately simulate diffuse interreflection.

For the purposes of radiosity, introduced to computer graphics by [14], surfaces are assumed to be Lambertian diffuse reflectors, i.e., to reflect light with equal intensity in all directions. Surfaces are subdivided into discrete patches for which the radiosity, or energy per unit area leaving the surface, is assumed to be constant. A radiosity equation then expresses the patch radiosity as the sum of energy contributions from all other patches:

$$B_i A_i = E_i A_i + \rho_i \sum_{j=1}^{n} B_j A_j F_{ji}$$

with B_i the radiosity of surface i (energy per unit area), A_i, A_j the area of surface i and j, E_i the emitted energy per unit area, ρ_i the reflectivity of surface i, and F_{ji} the *form-factor* from surface j to surface i giving the fraction of the energy leaving patch j that arrives at patch i.

ERIC

A radiosity equation exists for every patch in the environment. The resulting system of simultaneous equations represents the interchange of energy via interreflection and emission for the environment. Solving the system of equations gives the patch radiosities, which provide a discrete representation of the diffuse shading of the scene. Being view-independent (the shading of Lambertian surfaces does not depend on the view direction), the radiosity method is an object-space shading algorithm. Once the system of equations has been solved, the shading of all surfaces is known and it is possible to view the shaded scene from any eye position without recomputing the illumination. By passing the shaded patches to a graphics accelerator a shaded scene can be walked through at interactive rates.

An efficient solution for the treatment of shadows has been introduced by the *hemi-cube* algorithm [7]. The form-factors from a given patch are determined by projecting all other patches onto the five faces of a hemi-cube positioned above the given patch. Occlusion is taken into account by scan converting the projected polygons into a z-buffer for each face of the hemi-cube. The form-factor to a given patch can be determined by summing a contribution for each hemi-cube pixel in which the patch is visible.

The high computational complexity (form-factors must be computed from every patch to every other patch resulting in a time complexity of $O(n^2)$; computation and storage of all form-factors before solving the system of equations also requires $O(n^2)$ storage) was addressed by the progressive radiosity approach [6]. · the conventional *gathering* radiosity approach the radiosity equations are solved using Gauss-Seidel iteration. At each step the radiosity of a single patch is updated based on the current radiosities of all other patches, thus gathering illumination from all other patches into a single receiving patch.

In the progressive *shooting* radiosity approach, on the other hand, the solution repeatedly executes the following steps:

1. select patch with greatest reflected and/or emitted energy
2. compute form-factors from that patch to all other patches
3. based on form-factors, add contribution from source patch to radiosity of every other patch
4. for display purposes only add an estimated ambient term based on the total unshot radiosity to the path radiosity

At each step the illumination provided by a single patch is distributed (shot out) to all other patches in the scene. In the early steps the source patches chosen will be the light emitters. Subsequent steps will select secondary sources, starting with those surfaces that received the most light directly from the light sources. Since each solution step updates the radiosity of all surfaces, the increasingly accurate result can be displayed following each step. Together with the estimated radiosity of all surfaces which haven't received anything so far useful images can be produced very early in the solution process.

Adding specular reflection to radiosity is typically based on the observation that view-independent and view-dependent shading methods have complimentary strengths and weaknesses. Radiosity, a view-independent method, has the advantage for shading diffuse surfaces. Ray tracing, a view-dependent method, has the advantage for shading specular surfaces. In a two-pass algorithm (first described by [33]) the shading of diffuse surfaces is determined during a view-independent first pass, and the shading of specular surfaces is added during a view-dependent second pass.

A simple example of such an algorithm computes diffuse shading using a standard radiosity solution and adds specular shading during ray tracing of a particular view. Whenever a ray hits a diffuse patch, the diffuse intensity is determined by interpolating from the pre-computed patch radiosities. This value is then used by the ray tracer's shading equation in place of the normal diffuse component. Although this simple scheme can produce impressive results, it does not completely solve the general shading problem. It ignores the illumination of diffuse surfaces by light reflected or refracted by specular surfaces. To include this type of effect, the radiosity solution itself must be extended to account for the illumination of diffuse surfaces by light reflected via specular surfaces [27, 28, 25].

Recent developments in radiosity have been directed towards improving the quality and speed of the algorithm. Improved sampling methods are providing more accurate shading and parallel and hardware implementations are achieving very fast solution times.

The hemi-cube algorithm is a uniform sampling method, since it computes form-factors by testing visibility at the regular spaced hemi-cube pixels. Thus it can produce aliasing, particularly where small bright sources of light provide a great deal of illumination. To overcome the aliasing inherent in the hemi-cube algorithm, the z-buffer must be replaced by a more flexible method of determining patch visibility. Ray casting is one approach providing a basis for non-uniform, adaptive and stochastic sampling [34]. A different approach to improving the quality of from-factors for progressive radiosity is a combination of the hemi-cube algorithm and an analytic form-factor equation [1].

Although the ability to 'walk around' in static scenes using precomputed solutions to the radiosity equation is quite important for many applications, new approaches are necessary for scenes where the geometry is constantly changing. Existing solutions address the problem of automatic patch subdivison in areas of high intensity gradients like shadow boundaries [8]. View-dependent or importance-driven solutions compute the radiosity of directly visible surfaces to high accuracy, while surfaces with an indirect affect are computed to a lower accuracy according to their contribution [30].

While most hardware based speed improvements are variations of the z-buffer technique (i.e. currently available hardware graphics engines are used to perform the expensive hemi-cube computation), the Pixel-Planes 5 architecture takes a different approach of directly computing the form-factors using Nusselt's analog [13]. The resulting speed per renderer is approximately 100,000 hemi-cubes per second thus providing real-time radiosity.

Modeling and Animation

The increasing availability of computer graphics in general together with the dramatic change of the price/performance ratio has concealed the fact that important aspects of realistic 3D computer graphics still have to be considered as hard problems. One of the big challenges in that field is to make complex modeling accessible at a large scale (as well as to make simple modeling easy).

Artificial Life and L-Systems

Some of the most impressive examples for artificial life techniques to generate complex graphics modeling botanic forms are based on L-systems, introduced by A. Lindenmayer as a theoretical framework for studying the development of multi-cellular organisms [22]. Enhanced with geometric features, L-system based plant models are detailed enough to allow realistic visualization of plant structures by standard image synthesis techniques [12].

Experiments with artificial life could yield new approaches to computer animation. As an example, the behavior of flocking bats in the movie *Batman Returns* could be simulated based on a few simple rules [23]. Executing these rules for each bat, the aggregation of bats simulates a flocking behavior found in group of birds.

Another approach to artificial life is based on genetic algorithms. Parameter sets are mutated and/or mated and so are symbolic expressions to create a variety of complex 2D

BEST COPY AVAILABLE

ERIC

and 3D images/models [29].

Particle Systems

Particles are objects that have mass, position, and velocity, and respond to forces, but have no spatial extent. Because of their simplicity, particles belong to the easiest objects to simulate. By applying forces to each particle (e.g. according to aerodynamics [35]) it will exhibit velocity, momentum, and acceleration. Integrating applied forces over time (in accordance with Newton's Laws of Motion) the particle system can be animated properly reacting to forces such as gravity or obstacles. Further attributes of a particle could be color and age to control the appearance and lifetime of each particle.

In other approaches the particles interact through the use of spring and damping elements to produce deformable models which exhibit plasticity, and bending [16]. Based on ideas from molecular dynamics particle systems have been used to model liquids and to mimic the effects of heat on solids [32]. By incorporating material dependent inter-particle constraints particle systems can be used to simulate specific properties such as those found in the draping of cloth [4]. Recently introduced oriented particle systems overcome the tendency to form solids and prefer to form surfaces instead [31].

Digital Image Warping

The term describes methods to deform images to arbitrary shapes and serves as the basis for a variety of applications. In 'traditional' computer graphics *texture mapping* which is one form of image warping has been used for some time. With the Michael Jackson video "Black or White" *morphing* (derived from 'metamorphosis') [37, 2] has become a widely known and popular technique for animating 2D images. Having realized that the triangulation based deformation frequently used is just a special case of scattered data interpolation methods applied to deformation, current research focuses on the construction of suitable mapping functions [24].

Graphics Hardware

Graphics applications such as flight simulation, virtual reality, and computer-aided design demand high frame rates ($\geq$ 30 Hz) as well as high quality rendering. Future systems must be able to generate high-resolution images from large data sets (in the order of millions of primitives) with realistic rendering techniques such as P' ng shading, texturing, and antialiasing.

In order to achieve this goal architectures have to overcome two bottlenecks: floating-point performance for the processing of the scene's geometry and low bandwidth to the frame buffer. The need for parallel solutions becomes obvious with the following example [20]: the rendering of a scene containing 100,000 triangles (assuming an average size of 50 pixels) updated at 30 Hz requires approximately 350 MFLOPs for the geometry processing, 750 million integer operations for the rasterization, and 450 million frame buffer accesses (includes clearing a 1280 × 1024 screen).

General purpose CPUs are becoming fast enough and, as with Intel's i860 [15], will have graphics enhancements added so that a collection of such processors can provide high performance graphics including imaging, video, advanced rendering, etc. Specialized hardware will still be able to outperform such devices, but may be cost-effective only for high-performance real-time applications.

BEST COPY AVAILABLE

ERIC

After wireframe and shaded polygons graphics hardware has reached the stage of texture mapped and anti-aliased polygons at 'real-time' speed enabling the implementation of experimental user interfaces like *virtual reality*. Texture especially aids in surface perception and offers a powerful degree of freedom for data visualization.

Significant bandwidth improvements will be achieved when the rendering engine and pixel memory are merged on the same chip. In such a configuration, entire spans of a texture-mapped polygon might be rendered in parallel on a single chip [19], or entire polygons might be rendered as in Pixel Planes [13, 21].

The currently dominant architecture for display systems focuses on 'engines,' typically featuring a geometry processor whose output is fed to a raster processor. The components used to implement these two elements vary widely but the basic structure has remained unchanged over the last decade.

Recent architectures employ *object-parallelism* for geometry processing: primitives are distributed over a parallel array of processors, which perform modeling and viewing transformation and clipping. Rasterization is based on *image composition* where each rasterizer computes a full-screen image of a subset of the primitives. Based on visibility, these images are combined in logarithmic time to form the final image [20].

Additional gain in speed can be achieved by *deferred shading*, a general technique for speeding up the display process: calculations for complex shading models are postponed and only applied to the visible parts of the surfaces. Regardless how many surfaces 'cover' a single pixel, shading is performed just once [26].

Tasks which are not specialized or which require machines with extreme random-access bandwidth may be done on general-purpose massively-parallel multiprocessors. Advanced shading effects which are predominantly geometric (e.g. global illumination models) and path planning are typical examples. It is not clear whether graphics hardware will evolve to directly attack such global problems, or whether general-purpose massively-parallel processors will get there first.

Another open problem for designers of graphics hardware systems is the identification of new primitive elements of the next generation architectures which could be a ray, a voxel, a non-uniform rational B-Spline (NURBS) surface, or a solid texture.

If graphics hardware and algorithms continue to improve at current speed one could argue that a 'graphics building block' priced at $ 200 might become available within a few years [5]. Such a module would be based on RISC technology and multiple parallel data paths and consist of program memories, double-buffered true-color screen memories with z-buffer, and specialized hardware for real-time 2D and 3D high-end, textured graphics as well as for image processing, compression, and encryption. With a price tag like this almost every computer could be equiped with a graphics module of unprecedented power and functionality.

Software Architecture

3D application development has not yet become a mainstream technology for everyday application and user interface development. One reason for this limited growth is that the computational and rendering requirements of 3D are beyond the performance capabilities of most machines. The other major reason for the slow proliferation of 3D is that software libraries available today do not provide a programming model that is appropriate for widespread use by developers not familiar with 3D graphics programming. The latest generation of RISC-based workstations and personal computers are quite capable of meeting

BEST COPY AVAILABLE

the 3D performance challenge. As the hardware gets faster, software will become the critical factor in the further growth of 3D application development.

Currently available software libraries are of two types. *Hardware drawing libraries*, such as SGI's *GL*, HP's *Starbase*, and SUN's *XGL*, provide pixel and graphics primitive drawing commands as a software layer above hardware devices (typically a frame buffer). *Structured drawing libraries*, such as *GKS* and *PHIGS+* [17], provide structured drawing commands that are abstracted from the low level hardware interface.

All of these systems, however, are variants of display list technology. Designed to simplify the task of building applications using synthetic 3D graphics, these models imply that the same program organization and methods of user interaction are suitable for all graphics applications. While these techniques have served the graphics community well, they are being stretched by the size and complexity of today's applications. In addition, they appear to be inadequate for dealing with new issues such as multimedia, time-critical computing, and asynchronous user input.

The next generation of 3D software toolkits will be *object-based* rather than drawing based [36]. They will be composed of extensible sets of editable 3D objects that perform a variety of operations. Rendering will be one of the many operations that each objects implements. The 3D objects will be building blocks that lend themselves to programmer customization through techniques such as subclassing.

For example, to customize a ray tracer based on object-oriented design the application programmer only needs to know the basic functionality of a geometric object such a ray tracer is capable to render. In [11] the following methods (defined in virtual base class Object) have to be provided for each new object: bounding_vol() returns the bounding volume, intersect() computes the intersection between the object and a ray, check_intersect() just performs an intersection test, surface_normal() computes the normal to the surface at a given point, triangulate() appends a triangular approximation of the object to a triangle strip list.

A new object can be introduced by a) providing the class definition of the new object according to the above rules (i.e. deriving it from class Object) and b) by adding syntactic rules to the scene description grammar from which the (yacc based) parser of the scene description is generated. These two steps are sufficient for any new object. No other parts of the ray tracer have to be studied or even modified by the programmer.

Future object-oriented toolkits will blur the current distinctions between 2D, 3D, windows, and user interface programming. New object-oriented systems will treat all of these as similar objects that perform rendering, picking, event handling, animation, grouping, geometry computations, and other intrinsic functions. The programmer's task will be primarily focused on solving the application problems, rather than spending excessive amounts of time on the graphics and user interface implementation. Furthermore, creating 3D objects that have physical behavior will be much easier and thus more prevalent. Applications will rely less on indirect panel-driven user interfaces and instead provide direct manipulation 3D user interfaces that behave much like objects in the real world.

Another limitation of existing graphics application programmer's interfaces (APIs) is their inability to deal with multiple media and their synchronization as well as other aspects of time-critical and interactive computing. The high processing power and networking capabilities that will be available in affordable desktop machines in the near future will enable distributed applications characterized by much more powerful and effective human-machine interfaces. These new generation user interfaces will employ time-critical computation, present audible and visual data, and allow the time-critical sensing of human gestures for intimate interaction.

ERIC

Time-critical computing requires specifying the timeliness of computations to the operating system, which then must apply elaborate scheduling of resources to complete within a specified deadline. Furthermore, in cases of system overload, gracefully degrading the quality of a computation (obtaining only an approximation to the actual result), is acceptable in order to meet the deadline. In addition, it is essential to achieve time-critical computing in a general purpose, multi-tasking, and networked computer environment, and not in special purpose systems. Only in this way can we maintain compatibility with the existing body of software.

3D graphics systems are essentially synthetic media systems that make use of highly interactive and compact mathematical models, creating media streams. This is in contrast to presampled media systems, based on the reception, compression, transmission, and decompression of media streams. While time is implicit with presampled media, 3D graphics systems must incorporate explicit time modeling to achieve the synchronization and fusion between synthetic media and other media streams.

In next generation graphics systems ease of use will be a critical factor as interactive 3D is applied to new application areas. While advances in hardware is making interactive 3D graphics systems more affordable, the expense of developing applications which incorporate 3D is still quite high. One reason being that existing APIs require a high degree of graphics programming expertise to use. What is needed instead, is a foundation for interactive graphics that enables individuals who are not computer graphics sophisticates to easily incorporate dynamic 3D imagery into their applications — applications that may be highly interactive and imagery that is driven by the application data.

Standards

Historically, software developers with 3D graphics requirements could either base their applications on (proprietary) hardware drawing libraries or on (internationally standardized) structured drawing libraries. Application developers with a high performance requirement typically took the first approach — and absorbed the extra costs of porting and supporting the different platform's graphics environments. Customers for whom portability and cost were most important selected the international standard approach, but suffered in performance in some cases.

PEX is a 3D technology based on PHIGS. The X Consortium, the governing organization for PEX, defined *PEXlib*, an API for PEX offering a standardized approach for low level graphics rendering. PEXlib with approximately 800 functions promises high performance 3D immediate mode rendering in an X environment, something PHIGS has lacked!. A traditional criticism of the PHIGS and PEX standards is that individual companies have implemented vendor specific versions which made portability difficult.

Silicon Graphics, the company with the leading proprietary API, decided to license its IRIS Graphics Library to the Computer Industry. *OpenGL*, the publicly available version of this technology, has been rearchitected to become window system independent and thus vendor neutral. OpenGL is a rendering only API with all of the advanced graphics features from earlier releases. SGI has created a governing organization, the Architecture Review Board composed of PC and workstation companies, in an attempt to create an open platform for all licensees. In July 1992 the board released the first specification of OpenGL with approximately 200 functions

With SUN and HP favoring PEX and SGI and IBM promoting OpenGL it will be the software vendors' vote which API will be the more successful one.

ISO completed its work on first generation graphics standards with the standardization of PHIGS+. In response to the need for a coordinated method for addressing all aspects of multi-media objects ISO has started the work on second generation graphics standards with an initial draft for a Presentation Environment for Multi-Media Objects, called *PREMO* [18]. Within the context of PREMO a multi-media object consists of static or dynamic graphics, synthetic graphics of all types, audio, still images, moving images including video, or any other content type or combination of content types. The multi-part document intents to support a wide range of multi-media applications (from simple drawings up to full motion video) in a consistent way. Some parts are expected to reach IS status as early as 1996.

Bibliography

[1] BAUM D. R., RUSHMEIER H. E., WINGET J. M.: Improving radiosity solutions through the use of analytically determined form-factors. *Computer Graphics 23*, 3 (July 1989), 325–334.

[2] BEIER T., NEELY S.: Feature-based image metamorphosis. *Computer Graphics 26*, 2 (July 1992), 35–42.

[3] BLINN J. F.: Models of light reflection for computer synthesized pictures. *Computer Graphics 11*, 2 (July 1977), 192–198.

[4] BREEN D., HOUSE D., GETTO P.: A particle-based computational model. In *Scientific Visualization of Physical Phenomena*. Springer, 1991, pp. 113–131.

[5] CLARK J.: A TeleComputer. *Computer Graphics 26*, 2 (July 1992), 19–23.

[6] COHEN M. F., CHEN S. E., WALLACE J. R., GREENBERG D. P.: A progressive refinement approach to fast radiosity image generation. *Computer Graphics 22*, 4 (Aug. 1988), 75–84.

[7] COHEN M. F., GREENBERG D. P.: The Hemi-Cube: A radiosity solution for complex environments. *Computer Graphics 19*, 3 (July 1985), 31–40.

[8] COHEN M. F., GREENBERG D. P., IMMEL D. S., BROCK P. J.: An efficient radiosity approach for realistic image synthesis. *IEEE CG&A 6*, 3 (Mar. 1986), 25–35.

[9] COOK R. L., PORTER T., CARPENTER L.: Distributed ray tracing. *Computer Graphics 18*, 3 (July 1984), 137–145.

[10] COOK R. L., TORRANCE K. E.: A reflectance model for computer graphics. *ACM TOG 1*, 1 (Jan. 1982), 7–24.

[11] FELLNER D. W.: *Extensible Image Synthesis*. Tech. Rep. IAI-TR-93-4, University of Bonn, Dept. of Computer Science, Bonn, Germany, Mar. 1993.

[12] FOWLER D. R., PRUSINKIEWICZ P., BATTJES J.: A collision-based model of spiral phyllotaxis. *Computer Graphics 26*, 2 (July 1992), 361–368.

[13] FUCHS H., POULTON J., EYLES J., GREER T., GOLDFEATHER J., ELLSWORTH D., MOLNAR S., TURK G., TEBBS B., ISRAEL L.: Pixel-Planes 5: A heterogeneous multiprocessor graphics system using processor-enhanced memories. *Computer Graphics 23*, 3 (July 1989), 79–88.

[14] GORAL C. M., TORRANCE K. E., GREENBERG D. P., BATTAILE B.: Modeling the interaction of light between diffuse surfaces. *Computer Graphics 18*, 3 (July 1984), 213–222.

[15] GRIMES J., KOHN L., BHARADHWAJ R.: The Intel i860 64-bit processor: A general-purpose CPU with 3D graphics capabilities. *IEEE CG&A 9*, 4 (July 1989), 85–94.

[16] HAUMANN D., WEJCHERT J., ARYA K., BACON B., KHORASANI A., NORTON A., SWEENEY P.: An application of motion design and control for physically based animation. In *Graphics Interface '91* (June 1991), pp. 279–286.

[17] ISO: *Information Processing Systems – Computer Graphics – Programmer's Hierarchical Interactive Graphics System (PHIGS). Amendment 1–3*, IS 9592/Am. 1-3, Feb. 1992.

[18] ISO: *Information Processing Systems – Computer Graphics and Image Processing – Presentation Environment for Multimedia Objects (PREMO)*. Initial Draft, Nov. 1992.

[19] KELLEY M., WINNER S., GOULD K.: A scalable hardware render accelerator using a modified scanline algorithm. *Computer Graphics 26*, 2 (July 1992), 241–248.

[20] MOLNAR S., EYLES J., POULTON J.: PixelFlow: High-speed rendering using image composition. *Computer Graphics 26*, 2 (July 1992), 231–240.

[21] POTMESIL M., HOFFERT E. M.: The Pixel Machine: A parallel image computer. *Computer Graphics 23*, 3 (July 1989), 69–78.

[22] PRUSINKIEWICZ P., LINDENMAYER A.: *The Algorithmic Beauty of Plants*. Springer, 1990.

[23] REYNOLDS C. W.: Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics 21*, 4 (July 1987), 25–34.

[24] RUPRECHT D., MÜLLER H.: *Image Warping with Scattered Data Interpolation Methods*. Tech. Rep. 443, Dept. of Computer Science, LS VII, Univ. of Dortmund, Germany, Nov. 1992.

[25] RUSHMEIER H. E., TORRANCE K. E.: Extending the radiosity method to include specularly reflecting and tranlucent materials. *ACM TOG 9*, 1 (Jan. 1990), 1–27.

[26] SCHNEIDER B. O., CLAUSSEN U.: PROOF: An architecture for rendering in object-space. In *Advances in Computer Graphics Hardware III*, Kuijk A. A. M., (Ed.), Eurographic Seminars. Springer, 1991, pp. 121–140.

[27] SHAO M.-Z., PENG Q.-S., LIANG Y.-D.: A new radiosity approach by procedural refinements for realistic image synthesis. *Computer Graphics 22*, 4 (Aug. 1988), 93–99.

[28] SILLION F., PUECH C.: A general two-pass method integrating specular and diffuse reflection. *Computer Graphics 23*, 3 (July 1989), 335–344.

[29] SIMS K.: Artificial evolution for computer graphics. *Computer Graphics 25*, 4 (July 1991), 319–328.

[30] SMITS B. E., ARVO J. R., SALESIN D. H.: An importance-driven radiosity algorithm. *Computer Graphics 26*, 2 (July 1992), 273–282.

[31] SZELISKI R., TONNESEN D.: Surface modeling with oriented particle systems. *Computer Graphics 26*, 2 (July 1992), 185–194.

[32] TONNESEN D.: Modeling liquids and solids using thermal particles. In *Graphics Interface '91* (June 1991), pp. 255–262.

[33] WALLACE J. R., COHEN M. F., GREENBERG D. P.: A two-pass solution to the rendering equation: a synthesis of ray tracing and radiosity methods. *Computer Graphics 21*, 4 (July 1987), 311–320.

[34] WALLACE J. R., ELMQUIST K. A., HAINES E. A.: A ray tracing algorithm for progressive radiosity. *Computer Graphics 23*, 3 (July 1989), 315–324.

[35] WEJCHERT J., HAUMANN D.: Animation aerodynamics. *Computer Graphics 25*, 4 (July 1991), 19–22.

[36] WISSKIRCHEN P.: *Object-Oriented Graphics: From GKS and PHIGS to Object-Oriented Systems*. Springer, 1990.

[37] WOLBERG G.: *Digital Image Warping*. IEEE Computer Society Press, 1990.

ERIC

The IBM "Advanced Technology Classroom" Project: A Report on An Experimental, Multimedia, Interactive, Instructional System

MARLA J. FISCHER
Ancell School of Business
Department of Management Information Systems
Western Connecticut State University, Danbury, CT 06810, USA

Introduction

Advancements in computer and information technology are having a profound effect on the manner in which instruction is delivered in a growing number of university classrooms. Hi-tech, multimedia, instructional delivery systems are gradually eclipsing conventional methods of instruction and will soon replace the lecterns and blackboards that have bee.. the hallmark of university classrooms the world over for several centuries.

In the summer of 1988, as a result of a grant awarded by the IBM Corporation, Western Connecticut State University became one of the first universities to depart from past practice and formally make the transition to a "new-age" classroom[1]. Under the conditions of the grant, the university received an experimental, integrated, hi-tech electronic classroom that had been previously used exclusively for in-house training of corporate executives at IBM's Management Training Center in Armonk, New York. In return, the university field-tested the classroom's effectiveness, developed courseware compatible with the system, and served as a demonstration site for what many regard as a prototype for classrooms of the future.

In this presentation I will report on the IBM/WCSU "Advanced Technology Classroom" Project (cf. Fischer, 1993). In doing so, I will first describe the components and configuration of the system and point out what makes the system unique. Following that, I will indicate how this innovative instructional system capitalizes on established principles of learning. Finally, I will describe some of the research we have conducted examining the efficacy of the system.

Description of the Classroom

The "Advanced Technology Classroom" is an integrated, multimedia, interactive, instructional delivery system. In the ATC a large (4ft X 4ft), high-resolution video screen replaces the traditional blackboard. A specially designed electronic podium containing a central microcomputer serves as the master control panel for the classroom. From this podium, the instructor can control and coordinate all components of the system.

What makes the podium unique is its ability to orchestrate seamless, multimedia presentations without requiring the instructor to independently activate each component of the instructional sequence. The instructor can, for example, present information either on the electronic blackboard using a sonic pen or on the video screen where computer-generated graphics, videotape or material from laser disks can be used to illustrate key concepts and ideas. Without leaving the podium the instructor can, as an example, present introductory information from an audio cassette, follow this immediately by a videotape/laser disk segment illustrating the concept, and follow this, in turn, with instructor-prepared visuals and graphics that highlight the key points that were introduced. The instructor can access scanned images, full-motion video, digitized images, and computer-generated graphics from the podium as well as from a remote, hand-held unit.

Each student's desk in the "Advanced Technology Classroom" is equipped with a student response unit which can be used by students to respond to variable-choice formatted questions posed by the instructor. These student keypad units are not designed to take the place of normal

[1]IBM awarded the Advanced Technology Classroom Grant to Western Connecticut State University as the result of a proposal submitted by Drs. R. Benson, G. Buccini, and M. Fischer in April, 1988.

pupil-instructor interaction, but are used instead to keep the instructor apprised of how well each student is comprehending information during the course of the lesson. The microcomputer built into the podium informs the instructor of each student's response to the keypad questions and, in addition, provides summary data on the performance of the class as a whole. Access to these data allows the instructor to make "on-line" decisions regarding the pace of instruction; the instructor can ascertain when to move on to new information or when to provide further explication of topics not fully comprehended. This feature of the system eliminates the need for instructors to scan the faces of the class in search of nonverbal cues that might signal frustration or confusion.

The design of the ATC also allows the instructor to share the keypad question results with the class. The instructor can display on screen not only a histogram indicating what percentage of the class answered a particular keypad question correctly (or incorrectly), but also comparison data showing how the class compares to previous classes responding to the same question.

An additional feature of WestConn's ATC is that each student's desk contains a personal computer. The PC unit is mounted under a specially designed desk[2]. A glass top on the desk permits the student to view a recessed screen; a slide-out drawer contains the keyboard. The advantage of this design is that the visual field in the classroom is not obstructed by PC hardware that might otherwise prevent normal viewing and interaction. The desks in the ATC are arranged in a U-formation to facilitate high levels of interactivity. The PCs in the ATC are networked within the classroom (LAN) and connected to the university Mainframe providing both local and campus-wide connectivity.

Pedagogical Features of the System

There is a paradox in the fact that while learning theorists long ago identified the critical elements essential for facilitating learning (Gagne, 1970; Hilgard & Bower, 1966), these elements are often ignored in the design and delivery of instruction, particularly at the university level. Perhaps, it is because most faculty at the university level hold a Ph.D. degree and Ph.D. programs are not structured or intended to prepare teachers (Lewis & Altbach, 1992). For optimal learning to take place, there has to be a careful marriage between the structure of the lesson and established principles of learning. Perhaps the greatest virtue of the ATC is its capability for weaving basic principles of learning into the fabric of a lesson/course.

One example of the ATC's capability for capitalizing on established principles of learning is its provisions for interactivity (Horowitz, 1988). Basically, interactivity refers to the dynamic exchange that takes place between the learner and other components of the system; the extent to which the learner becomes an active participant in the learning process. Interactivity may include several kinds of engagement: a learner may engage the content/materials of the lesson, a learner may engage the instructor who is directing the lesson, or a learner may engage other students participating in the lesson. Each form of interactivity plays an important role in maximizing opportunities for learning.

The design of the "Advanced Technology Classroom" enhances opportunities for all forms of interactivity. In the ATC students interact with the instructor not only through the traditional mode of questions and answers exchanged during instruction, but also through the use of keypad response units and student PCs. The incorporation of periodic keypad questions into a lesson forces full participation by all members of the class and provides the instructor with a more accurate indication of student progress than could be attained through sampling that is based solely on voluntary responses. An inherent benefit of the keypad units is the confidentiality they provide to students, since responses to keypad questions are essentially confidential (only the instructor knows). This means that students can respond to questions without fear of being publicly embarrassed by an incorrect answer. In a conventional classroom a sizeable number of students withdraw from participation, in part, for this reason.

Interactivity with the content/materials, the instructor, and with other members of the class is also facilitated by the U-formation of the classroom desks and by the accessibility of personal computers. The personal computers play an especially important role in fostering interactivity. There are, for example, numerous software packages that enable students in math classes to analyze data

[2]The personal computers and special desks for the ATC were acquired through a State of Connecticut, Department of Higher Education, High Technology Program Grant awarded to M. Fischer in August, 1989.

ERIC

that the instructor has provided as part of a lesson. There are also software programs that enable students to use their PCs to type in their own thoughts or comments on a particular topic which can then be released by the student for inspection by the instructor or by the class-as-a-whole. Classroom PCs can be used for library searches and for a multitude of other instructional purposes during the course of a lesson.

A related pedagogical virtue of the ATC is that it allows students to monitor their own progress. The increased opportunities for interactivity provide students with a steady stream of feedback on the correctness or incorrectness of their responses. This immediate feedback serves to either reinforce prior learning or, alternatively, signal the student that more study and practice are necessary.

The ATC is also ideal for assisting the instructor in making the lesson relevant and meaningful. With its capability for utilizing multimedia, the ATC can make the learning experience dimensionally richer. With much greater facility than would be possible in a conventional college classroom, the ATC can open the door to a wide range of experiences. Students can be taken to distant locations around the globe to see "first-hand" the events and conditions they are studying. They can meet the experts in their discipline and hear personal accounts of their contributions. They can be exposed to broad applications of the principles and theories they are studying. In short, the learning experience can be made to transcend the classroom. By transcending the boundaries otherwise imposed by space and time, the ATC provides access to a larger body of information in ways not possible in a traditional classroom. The end result is an enriched experiential base for learning.

The ability of the ATC to present first-hand information is by no means trivial. Since the beginning of time, instruction has centered heavily on information handed down from an indeterminable number of sources. The person acting as instructor served both as the repository for information and the conduit through which that information was communicated to the student. In serving in this dual role, the instructor (not the information) was the focus of the lesson, as well as the interface through which the learner came in contact with information. The depth and purity of the information presented rested entirely on the accuracy of the instructor's sources (whatever they might be) and on the instructor's ability to communicate the information with minimal distortion. With instructional systems such as the ATC, we move closer than ever before to the purest form of information transfer, to instruction that is first-hand and experiential, as opposed to second-hand and anecdotal.

Yet another pedagogical virtue of the system is its capability for catering to individual differences in learning styles. It accomplishes this in a number of ways. First, the very nature of the system encourages the instructor to present information both auditorially and visually. Though it is customary for instructors in traditional classrooms to periodically back-up their spoken lectures with a variety of visuals, there is a tendency to rely primarily on the auditory mode of presenting information. The design and operation of the ATC tends to strike a better balance in how information is communicated. In doing so, the ATC manages to accommodate both students who are auditory learners as well as those who learn best when information is presented visually. Indeed, there is a body of literature suggesting that for all students, learning is facilitated when abstract concepts are visually broken down into more concrete, digestible units (Arnheim, 1969). Through what has been termed "conceptual visualization" the learner comes to more readily apprehend concepts that would remain unclear in the absence of visual dissection.

A second way in which the system meets the needs of students with different learning styles is its capability for accommodating both extraverted and introverted learners. While extraverted learners tend to be more comfortable participating in classroom interactions, their introverted counterparts are usually more reluctant to engage either the teacher or their fellow classmates (Eysenck, 1976). The incorporation of the keypad units into the ATC affords an opportunity for introverted students to participate covertly in the lesson without having to sacrifice whatever need they might have for anonymity.

A final pedagogical benefit of the ATC, and one that is extremely important, is its impact on the structure and organization of the lesson. Since the use of the ATC requires pre-programming the system for the presentation of instructional materials, the instructor is obliged to pay strict attention to the structure and orchestration of each lesson. As a consequence, instructors using the ATC tend to have a greater awareness of the structural integrity of their lessons that those whose presentation format allows them to be more casual about how course content will be presented.

Student and Faculty Reactions to the System

Given the many pedagogical features of the ATC, the question remains, is it effective? To answer this question faculty participating in the ATC project at WestConn administered a survey to students taking courses in the "Advanced Technology Classroom[3]." The survey asked students to evaluate specific features of the ATC and to rate the overall effectiveness of the ATC as compared to a more traditional classroom setting. The features that were examined included: the effectiveness of the computer-based visuals, the impact of the ATC on the organization and sequencing of instruction, the effectiveness of the keypad response units, and the effectiveness of the ATC in fostering interest and motivation.

Students were asked to respond to each survey item by indicating their rating on a 7-point Likert scale. Three points on the scale were pre-labeled to assist students in arriving at their decisions. A low rating of "1" indicated that, for the feature under consideration, instruction in the ATC was "much less effective" as compared to a regular classroom. A rating of "4", the mid-point on the scale, indicated that the two instructional environments were perceived as being "the same." A high rating of "7", indicated that the ATC was perceived to be "much better" with regard to the feature under consideration.

The results of the survey indicated a high degree of student satisfaction with the ATC. For each feature examined, 90 percent (or higher) of the students rated the ATC as being better than the traditional classroom with the majority of responses clustering around "6" or "7" on the seven-point scale. Anecdotal remarks from students included comments such as: "sparks interest", "sticks in your mind", "keeps all students involved", "allows maximum participation", and "shows us things we wouldn't see otherwise."

Like students, instructors teaching in the ATC also had positive reactions about the instructional system. All of the instructors involved with the ATC project felt that the electronic classroom stimulated student-learning. They reported increased attentiveness from their students and a greater number of unsolicited statements extolling the virtues of the classroom. Instructors reported, in addition, that the system enhanced their instructional effectiveness by promoting greater classroom efficiency. They found less of a need to repeat information and received fewer requests from students for clarifying examples. The time that would otherwise have been allocated for these activities was therefore available to use for other purposes, such as providing additional instruction or engaging in other class activities. Although the time savings reported varied across instructors, it ranged from 5 to 15 minutes of additional time per class period.

Indeed, the ATC was so well received by students, faculty, and administration that it served as a catalyst in initiating campus-wide interest in information technology systems and provided the impetus for the installation of additional hi-tech classrooms at the university.

[3]The results reported here are based on data collected by Drs. G. Buccini, J. Cilliza, and M. Fischer.

References

Arnheim, R. (1969). *Visual thinking.* Berkeley, CA: University of California Press.

Eysenck, H. J. (1976). *The measurement of personality.* Baltimore, MD: University Park Press.

Fischer, M. J. (1993). The advanced technology classroom. In L. Lloyd (Ed.), *Technology in higher education: Case studies on the use of computers, networks, and multimedia.* Westport, CT: Meckler Press.

Gagne, R. M. (1979). *The conditions of learning* (2nd Ed.). New York: Holt, Rinehart & Winston.

Hilgard, E. & Bower, G. H. (1966). *Theories of learning.* New York: Appleton-Century Crofts.

Horowitz, H. (1988). Student response systems: Interactivity in a classroom environment. Paper presented at the Sixth Conference of Interactive Instructional Delivery, Society of Applied Learning.

Lewis, L. & Altbach, P. (1992). The new civil rights law and doctoral education. *Academe*, May/June, 12-14.

Interactive Information Center

DON FORESTA[1] , JOHANN GÖTSCHL[2],
HERMANN MAURER[3], WOLFGANG SCHINAGL[4]
[1]*Department of Video Art at the Ecole Nationale Supérieure des Arts Décoratifs, Paris and Department of Information Science at Paris University II, I.M.E. International Media Exchange and I.M.C. sarl International Media Consultants, Paris, France*
[2]*Ludwig Boltzmann Institute for Science of Research, Graz and Department of Philosophy, University of Graz, A-8010 Graz, Austria*
[3]*Institutes for Information Processing, Graz (IIG) and Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology and Institute for Multi-Media Information Systems of Joanneum Research Graz, A-8010 Graz, Austria*
[4]*Ludwig Boltzmann Institute for Science of Research, Graz and Department of Applied Computer Science and Computer-Systems Research of the Economic Promotion Institute of the Chamber of Commerce of Styria, A-8010 Graz, Austria*

Abstract

This document describes what we call an *Interactive Information Center* (IIC). An evaluation of both cultural and economic aspects has led us to believe that a major requirement of our modern industrial societies, i.e. the rapid dissemination of knowledge and skills, can no longer be met by educational institutions which exclusively rely on classical training schemes, even if modern technologies are used (i.e. computer support).

In our opinion training should be preceded by the use of new infrastructures, which interactively illustrate the purpose knowledge and skill have in our society. We are referring to new infrastructures for motivation, games, experiment, simulation, counseling etc. The Interactive Information Center has been devised for the purpose of encouraging people to search for suitable information methodically (e.g. training and education), stimulating interaction with simulations and having advisory dialogues with people or machines. Finally, it is a guide to a modern specialized, yet holistic training programme.

1. Introduction: A New Diagnosis: Categories for a New Modelling of Socio-cultural Systems

"Modern societies are characterized by their continuous endeavours to improve and renew the basic structures needed for the evolutionary progress of all members of the community. In their ideal form they represent the unity of education and training, i.e. the unity of rational knowledge, technical skill and humane attitude. Only in the openness and dynamic development of modern societies can opportunities arise to realize the highest goals: to contribute to the development of and connection between cultural and material growth." Johann Götschl

209

20th century societies may be characterized by technological and social developments, which led to a new understanding of rationality (Götschl, 1992). Some reasons for the reformulation and transformation of the concept of rationality are:

a) the increase of dynamics and complexity (connectivity) of systems
b) the understanding of the limits of regulation and control (self-organizational processes in physical, biological and socio-cultural systems)
c) the understanding of the limits of predictability (e.g. chaos-theory)
d) the understanding of the necessity of the increase of justification and evaluation and transparency
e) the understanding of the necessity of new forms of decision-making
f) the understanding of the correspondence of the increase of knowledge and the increase of competence of decision-makers

The consequences of a new concept of rationality represent a new *world-view*. A new *world-view* is necessary because it possibly offers new solutions to the various problems of nature and mankind (e.g. ecological, socio-cultural, multi-cultural, economic, political, scientific, existential problems).

The question is how we construct or generate initial conditions for such an adequate *world-view*, adequate to the various problems around us.

Technological and scientific progress in our society moves so fast that an individual with an "ordinary" *world-view* has a rather reduced chance to locate and "culturalize" his or her position between the dynamics of nature, society, technology and the emergence of humanity.

This leads to a deficiency of self-determination of (wo)man, which in turn leads to estrangement, ignorance and chaotic behavior (Postman, 1992).

In order to use and transform the categorial changes of the world it is necessary to create a new infrastructure, which enhances the process of diffusion of scientific and technological knowledge into all levels of society. We suggest therefore to establish *Interactive Information Centers* in connection with existing institutions of higher education and training.

We feel there is an urgent need for new concepts of and instruments for the dissemination of information, because the "half-decay-rate" of knowledge has been reduced to less than six years in key areas of knowledge, e.g. biology, chemistry, medicine and information and computer science. This implies that even specialists who are unwilling or unable to keep up their studies in their own fields of knowledge have but 25% of the information available in this field after about twelve years.

In the field of computer science the development cycle of new hardware has been reduced to a mere 18 months. Because of the computerization of all dimensions of life it is evident that the dynamics of knowledge-transformation will further increase.

Therefore, it is necessary to develop new and efficient methods for the dissemination of knowledge, both in the form of specific training programs and in the form of "lifelong study". This situation is complicated by the fact that existing professions increasingly change their profile and that new professions emerge. Moreover, contradictory demands are made on learners as both expert training and holistic integration of knowledge are required.

Modern industrial societies thus face four major challenges:

1. There is an urgent need for the improvement of the dissemination of information.
2. There is a growing demand for information retrieval and/or the retrieval of structured information (knowledge).
3. The restructuring of existing professions and the emergence of new professions should be supported by suitable institutions of training.
4. Social developments should not be seen as isolated events but as being interdependent processes. The answer is neither "universal knowledge" nor "specialized expertise". A new type of expert training is required: one that generates understanding of general aspects and

ERIC

interdependencies and at the same time imparts methods for knowledge acquisition management to help acquire knowledge quickly and according to the level of one's needs.

2. A New Conceptional and Instrumental Reaction: The Interactive Information Center and Its Three Components: Motivation, Simulation and Training

As a possible answer to these challenges, we suggest the establishment of an *Interactive Information Center* (IIC). It consists of three components: a *Motivation Center* (MC), which provides the trainee with information in a lively and entertaining way ("edutainment", "infotainment") and arouses the desire to acquire further knowledge; a *Simulation Center* (SC), which enables the user to experiment with the virtual images of real processes of life and imparts insight by offering him advise, counseling and special information; and a *Training Center* (TC), which provides intense, continuously up-dated, in-depth training schemes for the participants, who are well prepared and motivated.

In the following three sections (3-5) the three sub-centers, *Motivation Center*, *Simulation Center* and *Training Center,* will be discussed separately. In sections 6 and 7 we will summarize the way the three sub-centers are coordinated and suggest further steps to be taken.

3. The Motivation Center

The *Motivation Center* (MC) is based upon the imperative of a "person-centered approach". One might simplify matters and say that the *Motivation Center* brings together people by means of high tech environment in a playful and entertaining fashion. People who are visiting the MC learn how to communicate in a new qualitative dimension. Only in the process of communication do motivational structures evolve, see (Götschl, 1992, 1990, 1987). Therefore it is necessary to start a highly individualized communication process by means of computer animations, computer simulations, interactive art objects, virtual reality environments, see (Maurer & Carlson, 1992; Hattinger et al, 1990) etc. The MC should grasp the people's focus of attention and provoke them to communicate and play. The communicational and game-like interactions should initiate self-organizing motivational structures for more innovative life-dynamics and life-styles, see (Gerken, 1990), which should result in innovative economic potentials for finding new products and new services. The high-tech scenario offers open motivational structures for initial and further training.

Computer-oriented infrastructures such as the *Electronic CAFE* (Communication Access for Everybody), see (Barry, 1991; Foresta, 1991; Maupas, 1992), the *Viewseum*, see (Maurer & Williams, 1991), *The Networked Virtual Art Museum*, see (Loeffler, 1992) or the *Ars Electronica Center*, see (Leopoldseder et al, 1993) are based on networked services such as the transmitting of high-quality digital information over ISDN telephone lines. These networks may realize long distance, multi-cultural and multi-user virtual reality environments, which may generate new qualities of psychological counseling, training and education, information on different occupations and career prospects in different countries, general information on the latest developments in the international job market, etc. One of the main components of the *Motivation Center* in terms of infrastructure is a substantial hypermedia system (HS). It provides comprehensive information in the form of video clips, images, encyclopedic data, diagrams and cartographic data on all aspects of modern life etc. The main advantage of such a hypermedia system is that it is so easy to handle that even newcomers to computing can cope. It is fun to work with, and thus creates and enhances motivation, see (Maurer, 1992) and (Maurer et al, 1992).

The communication facilities of the hypermedia system permit an extensive exchange of ideas and access to even remote data, paving the way for more advanced uses (see section 4).

4. The Simulation Center

The *Simulation Center* (SC) is based upon the imperative of a computer-supported processing of "world-information" which is relevant in cultural and economic terms. This imperative is put into practice by the following elements:

a) Application of high-performance PC networks which permit access to various data bases and thus make the local processing of data possible. This function may be part of the hypermedia system outlined in section 3. This "orthogonal" application enables the user of the MC to proceed to the SC easily.
b) Advice and recommendations for the user given both by people and by computers. Computer-supported counseling might be carried out by means of a computer conference with hypermedia systems (HS) and virtual reality systems (VRS).
c) Formation of models of real economic processes by using computer simulations or by means of cooperative "games" along the lines of VRS, where the computer has a guiding function.

5. The Training Center

The Training Center (TC) is based upon the idea that in the shortest possible time the trainees should reach a satisfactory level of training. This would be a level that suits his or her job situation and/or the general short-term and medium-term labour market conditions in the best possible manner. This idea might be realized by the following elements:

a) Dynamic planning of training in the field of applied information science
b) Holistic fashioning of training contents
c) Dynamic cooperation between the IIC and the economic sector
d) Realization of a)-c) by means of special training programs (training, courses, seminars, tutorials etc.) with HS and VRS as general infrastructure for group discussion, electronic library, etc.

The trainees use the whole networked infrastructure of the IIC. Training rooms are designed as "classrooms of the future", see (Maurer & Williams, 1991): About 9-12 trainees are sitting around an oval desk with built-in computer-terminals which are connected to a HS.

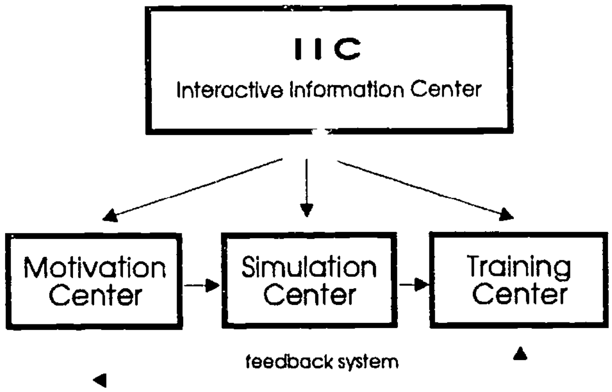
Explanations of the trainer are visualized at the terminal-display. Trainees have permanently local and remote access to encyclopedic data, various forms of dictionaries, a global bulletin board system, etc. and are connected to other infrastructures like the IIC.

6. The Holistic Integration of the Three Components

The idea of the IIC is based upon the insight that the traditional and classical training centers do no longer suffice to make the trainee experience the current and realize the future developments of modern society and put this knowledge into practice in his or her professional life. Our society is developing and changing rapidly, and its complexity is increasing. Therefore, it has become necessary to create the right conditions for modern education before training.

These conditions are: motivation of the individual to acquire more knowledge and skill through interactive technologies and simulations. The *Motivation Center* opens the trainee's eyes to the possible applications of computers, increases his or her interest in working with the computer and prepares him or her for the development of his or her personal interests in the *Simulation Center*. In a similar way the *Simulation Center* prepares the user for the core of the IIC, the *Training Center*.

Thus, the IIC can also be seen as a kind of channel in which a continuous diffusion of information of various kinds takes place. By means of the playful handling of high tech, this channel leads the visitors and users of the IIC from the motivation component (MC) to the simulation component (SC), where complex ideas may be tentatively tested with the highest degree of rationality and precision in HS and VRS. When these simulations reach a level of quality close to what can really be imagined, the trainee qualifies for specific training in the *Training Center* (TC). This process may be illustrated as follows: see attached diagram.



7. Outlook

With the IIC's key intention in mind we plan to carry out a comparative analysis of several international activities in the constructing of new hypermedia and virtual reality infrastructures, such as the Ars Electronica Center in Linz/Austria (start of construction in 1993), the Electronic CAFE in Paris/France (in operation), the Networked Virtual Art Museum in Pittsburgh/USA (planned), the Interactive Communication Center in Japan (planned), and we are working with the emerging technology of virtual reality and cyberspace systems in combination with hypermedia systems. The authors are involved in several such projects. The concept of the Interactive Information Center (planned, Graz/Austria) is the conclusion of a networked cultural-economic approach.

At this moment we are in the process of installing an international high-level *Advisory Board.* In a first IIC conference the members of the *Advisory Board* will communicate their ideas about the IIC and its subcomponents. Plans for the conference will be made after the members of the *Advisory Board* have been nominated. At the same time we are setting up an *Executive Council* for operational purposes, which will see to the local realization of the IIC.

8. References

Barry, J. (1991). Arts and Access and Coffee, Too. Electronic Café Lets Artists Interface Internationally. *International Herald Tribune, October 8, 1991,* 10-11

Foresta, D. (1991). The Many Worlds of Art, Science and the New Technologies. *LEONARDO, Vol. 24, No. 2,* 139-144

Foresta, D. (1991). Mondes Multiples. *Édition BàS avec le concours de la F.E.M.I.S.*

Gerken, G. (1990). Abschied vom Marketing. Interfusion statt Marketing. Econ Verlag, Düsseldorf, Wien, New York.

Götschl, J. (1992). Das wissenschaftlich-rationale Denken. Elemente auf dem Wege zu einer Wissenschaftskultur. In: Mantl [Hrsg.]: Politik in Österreich. Die Zweite Republik: Bestand und Wandel. Böhlau Verlag, Wien, Köln, Graz, 645-660

Götschl, J. (1990). Zur philosophischen Bedeutung des Paradigmas der Selbstorganisation für den Zusammenhang von Naturverständnis und Selbstverständnis. In: Krohn, W., Küppers, G. [Hrsg.]: Selbstorganisation. Aspekte einer wissenschaftlichen Revolution. Braunschweig/Wiesbaden, Vieweg, 181-202

Götschl, J. (1987). Zum Subjekt-Objekt-Problem von transzendentaler und evolutionärer Erkenntnistheorie. Elemente einer Konvergenzvermutung von transzendentaler und evolutionärer Erkenntnistheorien. *In: Lütterfelds, W. [Hrsg.]: Transzendentale oder evolutionäre Erkenntnistheorie.* Darmstadt, Wiss. Buchges., 285-306

Götschl, J. (1992). The Study of Nature and the Emergence of World View - Philosophical Reflections on E. Schrödinger's Approach. In: Götschl, J. [Ed]: Erwin Schrödinger's World View. The Dynamics of Knowledge and Reality; Theory and Decision Library. Series A: Philosophy and Methodology of the Social Sciences; Dordrecht, Kluwer Academic Publishers, 173-182

Hattinger, G., Russel, M., Schöpf, C., Weibel, P. [Hg.] (1990). Ars Electronica 1990: Band II: Virtuelle Welten. Veritas-Verlag Linz.

Leopoldseder, H., Janko, S., Königstorfer, T., Schöpf, C. (1993). AEC, Ars Electronica Center, Projektstudie. Zusammenfassende Darstellung der Projektstudie auf Basis des ART + COM Konzeptes, der Marketingstudie und der Einzelstudien, AEC-Verein, c/o ORF andesstudio Oberösterreich, A-4010 Linz, Franckstr. 2a, March 1993

Loeffler, C. E. (1992). The Networked Virtual Art Museum. Project at the Carnegie Mellon University, Pittsburgh, Department of Telecommunications and Virtual Reality

Maurer, H. & Williams, M. (1991). Hypermedia Systems and Other Computer Support as Infrastructure for Museums. *J. MCA 14*, 117-137

Maurer, H. (1992). Why Hypermedia Systems are Important. *ICCAL '92 Proc. LNCS 602*, Springer, 1-15

Maurer, H., Kappe, F., Scherbakov, N. (1992). Hyper PC - A New Training Tool and its Integration in a Large Hypermedia System. *ETTE '92* Paris.

Maurer, H. & Carlson, P. (1992). Computer Visualization, A Missing Organ, and Cyber-Equivalency. *Collegiate Microcomputer, Volume X Number 2*, May 1992

Maupas, P. (1992). Réseau numérique. Les artistes partagent leurs écrans. *01 INFORMATIQUE, Nr. 1200*, 21 Février 1992, p. 29

Postman, N. (1992). Das Technopol. Die Macht der Technologien und die Entmündigung der Gesellschaft; Frankfurt am Main, S. Fischer.

A Hypertext-Based Approach to Computer Science Education Unifying Programming Principles

WENDY A. L. FOWLER and RICHARD H. FOWLER
Department of Mathematics and Computer Science
University of Texas - Pan American, Edinburg, TX 78539, USA
Internet: fowler @ panam1.panam.edu

Introduction

Introductory computer science courses must provide students both a theoretical foundation for study of the discipline and more concrete skills necessary to begin implementing programs. To achieve these goals, students must learn general problem solving skills, algorithm design, and an implementation language. Additionally, software engineering and program design techques should be learned from the outset, and the beginning student must learn the basic software tools for programming and design. Maintaining a balance and perspective among these somewhat disparate skills is one of the principal challenges of the introductory courses.

A number of studies have investigated novice programming in general and the more specific issues of teaching programming in a computer science curriculum (Baile, 1991; Soloway & Spohrer, 1989; Koffman, Miller & Wardle, 1984). The Association for Computing Machinery's *Computing Curricula 1991* (Tucker, 1990) points out that programming encompasses all of the activities involved in the description, development, and implementation of algorithmic solutions to problems. The difficulty inherent in teaching program and algorithm design at the same time as program implementation is central to the issues of teaching programming. This difficulty is exacerbated by introductory computer science courses and texts that structure the presentation of design and programming principles around the introduction of a particular programming language. It is difficult to avoid this structuring, but for the student learning his or her first programming language there is an understandable tendency to loose sight of abstract concepts when trying to implement an algorithm in a particular language.

The challenge of teaching programming in the introductory courses lies in *simultaneously* teaching 1) general problem solving skills, 2) algorithm design, 3) program design, 4) a programming language in which to implement algorithms as programs, and 5) software tool use that supports design and implementation. Attempts to separate the teaching of these skills have typically focused on separating program design from the other facets. In general, this has resulted in students' acquiring better design skills (Baile, 1991).

The separation of program design and description from implementation is one of goals of computer aided software engineering (CASE) (Boehm, 1981). This suggests that commercially available CASE tools might play a role in the introductory computer science courses (Sidbury, Plishka & Beidler, 1989). Yet, commercially available CASE tools have some potential drawbacks: complexity of the CASE environment, an interaction style possibly different from other elements of the programming environment, documentation that is not appropriate for introductory students (Mynatt & Leventhal, 1990), and expense for both the software itself and the supporting hardware (Kiper, Lutz & Etlinger, 1992).

Student oriented software tools have been implemented that overcome some of these drawbacks. In a system designed for the beginning student the implicit enforcement of design methodology characteristic of CASE systems can be more closely matched to the needs of the student. Carrasquel, Roberts, and Pane (1989) describe a visually based system designed for students in introductory courses. The system allows students to enter and edit a structured design that specifies data and control flow among modules. The system can also interact with other programs for data display and code entry. Schweitzer and Teel (1989) developed a system for teaching structured design that automates several aspects of design and code generation. The SODA system (Hohmann, Guzdial, & Soloway, 1992) closely ties students to a computer aided design (CAD) model of program design and development. Programs that support student's design and implementation tasks in the introductory courses are a natural extension of a tool based approach automating some aspects of programming.

An Environment Integrating Design, Programming Language, and Hypertext

The environment we are using for our introductory programming courses provides students with a closely integrated suite of pedagogically oriented programs. The programs and the interrelationships among programs are designed to support the independent development of each of the skills necessary for programming, while keeping more abstract concepts of computer science in sight. The environment integrates design and programming tools with a hypertext of lecture material and laboratory exercises.

Figure 1 below shows the principal components of the environment. Design Tool, in the upper left corner, is a student oriented CASE tool focusing on visually based structured program design. It serves as a vehicle to develop skills in problem solving by problem decomposition and incorporates a pseudocode language designed for the beginning student.

The hypertext notebook is shown in the lower right of the screen. It includes all material presented in classroom lectures organized as hypertext. The notebook is used concurrently with Design Tool and the programming language environment and incorporates the materials used in student laboratory exercises. Each student has his or her own copy that can be annotated. The Pascal programming environment is shown in the upper right. Information can be easily transferred among each component of the environment.

Design Tool

Design Tool provides a visually based system for problem solving using problem decomposition, program design via structure charts, data flow checking, Pascal code generation, and report production. To facilitate separating the tasks students must master in learning to program, students' first experiences with the laboratory's environment are a series of exercises using Design Tool in problem decomposition. In completing design exercises, students are introduced to the diagramming and report writing facilities of Design Tool, as well as the general windowing and editing environment, before learning the Pascal language facilities.

Design Tool provides general tree display facilities for module creation and deletion, reordering and repositioning of modules, collapsing and expanding subtrees and windows, and various layouts of the tree structure. As shown in Figure 1, an overview diagram of the entire structure is always in view orienting the displayed modules to the complete program design. The overview also provides navigation facilities for moving within the module structure. The display facilities for panning, using the overview, and zooming in and out on parts of the structure are designed to facilitate the student's keeping track of the overall design during development.

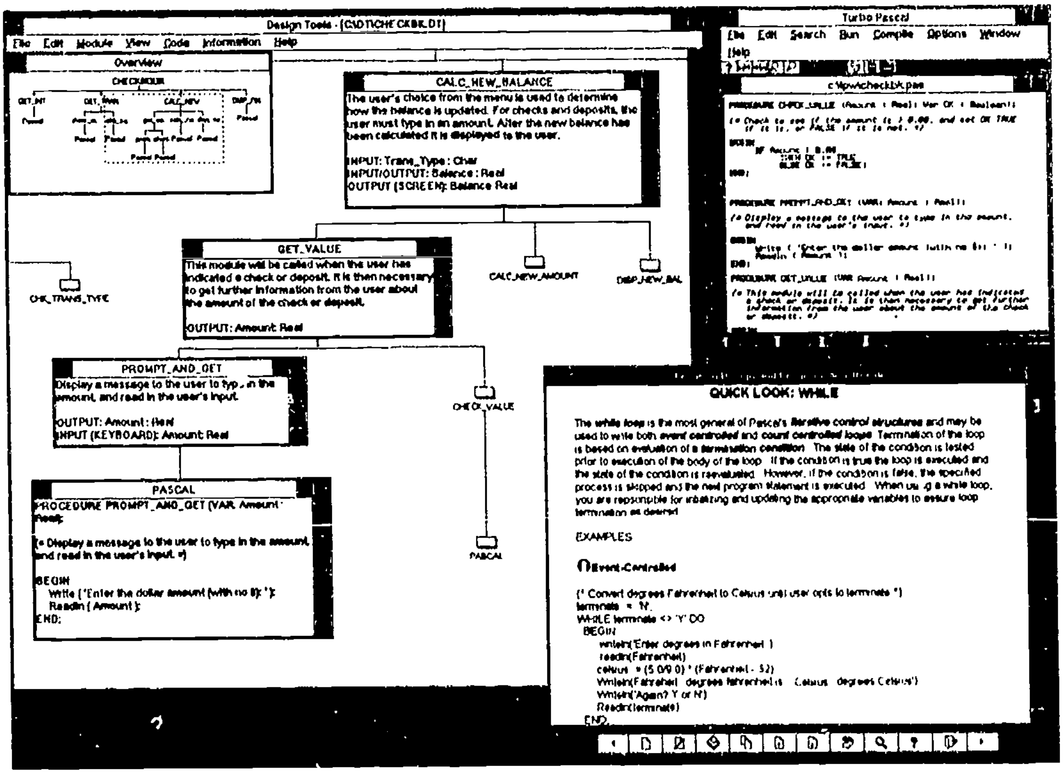


Figure 1. A typical screen showing the components of the student environment. The upper left window is the student oriented program design and problem solving tool showing an overview of the complete design and the student's arrangement of the design and code. The upper right window is the Pascal editor with code transferred from Design Tool. The hypertext in the lower right is opened to a section giving a tutorial overview of a programming construct.

A student ori· ·ted pseudocode language, centering on data flow and consistency, is available in Design Tool. A complete Pascal program module can be generated automatically from the pseudocode. Default values for language elements allow the student to design moderately complex problem solutions and generate most of the Pascal code very early. These early exercises serve to reinforce the lesson that programming focuses on problem solution and design and not the particular programming language or hardware.

CASE software is typically designed to tie its user, more or less rigidly, to a particular design methodology (Vessey, Jarvenpaa & Tractinsky, 1992). In Design Tool several constraint and consistency checking options can be used to enforce a particular methodology. For example, during the early weeks of the course, modules can only be created using a top down methodology. The student is required to start at the top of the module hierarchy, only adding modules di-

BEST COPY AVAILABLE

rectly below existing modules. Relaxing various constraints enforced by the system, regarding the availability and consistency of data, gives the user more freedom in the design of programs. With no module creation or data flow constraints, the system can be used as a sketchpad in creating and connecting modules. Data flow and consistency for module sets can be checked at the request of the user. Finally, the student's design created in Design Tool can be printed and used as his or her written design assignment. Subsequently, the design is used as the basis of the implementation. The information in each module can be copied directly into the program file when code is generated, serving as documentation and a guide for the developing program.

Hypertext Notebook

The hypertext serves to integrate concepts introduced in the classroom and text with the design and programming environment. The notebook contains the course lecture material and supplements: course syllabus, schedule, laboratory exercises, programming assignments, and a large set of designs and example programs. The system in which the hypertext is implemented allows the direct transfer of information to both Design Tool and the programming environment. Laboratory exercises are based on transferring programs and designs from the notebook to the programming environment. Students are required to exploit the integrated nature of the learning environment by using Design Tool to interactively examine program structure by executing example programs. Students are also encouraged to annotate their copies of the hypertext for study.

The hypertext is structured to reflect the sequence and scope of lectures. General lecture topics serve to organize the text by unit. Within units, main lecture points serve as organization points from which students can explore concepts in greater detail. Each unit provides numerous examples illustrating newly introduced concepts that build on previous examples and concepts. Students can experiment with the programs by copying them directly into the programming environment, and then compiling and executing the code. All examples used in the lecture are in the hypertext so that during lecture students can focus on a program's explanation and discussion and later use the hypertext to review and annotate le.

One element of the hypertext is Quick Look scr , as shown in Figure 1. A Quick Look provides a brief explication of a concept introduced in class and the text. The hypertext includes a series of these for the control and data structures presented in the course. The Quick Look screens provide a readily available and easy to use help facility for programming language semantics and data structure concepts analogous to the programming language syntax help facilities included in most programming language environments. These screens can be displayed together with corresponding laboratory exercises to help students gain a greater understanding of the implementation of concepts. Key concepts on the Quick Look screens are linked to more detailed information in the hypertext.

The development of the hypertext was in part motivated by a curriculum restructuring following the suggestions of the Association for Computing Machinery's *Curricula '91*. During the course of authoring the hypertext, frequent referral to *Curricula '91* suggested that for our own use it might be useful to have that document on-line. To meet this need, the printed document was scanned and converted to hypertext. We maintain a separate instructor's copy of the course hypertext that incorporates links to the *Curricula '91* hypertext.

Discussion

The structure of our introductory courses, based on an environment using a hypertext to integrate skills, is designed to address the challenges faced by beginning computer science students in developing programming expertise. The goal of the environment is to facilitate a pedagogy that allows instruction to focus on each of a set of somewhat disparate skills. The early introduction of program development tools as an integral part of program implementation serves as a foundation for a tool based approach used in more complex software development environments.

Design Tool is the first program students interact with, and the first weeks of the course focus on developing problem solving skills using problem decomposition. The use of this visually based environment for problem solving together with the hypertext is typically greeted with an enthusiasm that can help overcome some of the natural frustrations in learning any new software system. The repository of program designs in the hypertext notebook provides examples of complete designs for study in the early weeks. The first exposure to the programming language facilities entails transferring and executing programs from the student's hypertext notebook. This is designed to attenuate some of the difficulty in developing skills using the language facility, a "training wheels" approach.

The code generation facilities of Design Tool are designed to allow students to focus on algorithm development. Much of the potential for errors in data flow among modules is eliminated by constraint checking in Design Tool before moving to the implementation facilities. As students progress to the development of more complex designs, the strictly top down constraints of problem solving and module development are relaxed. This allows a design methodology reflecting both top-down and bottom-up program development appropriate for module reuse and a different design methodology. Finally, the hypertext notebook serves as a facility to physically and conceptually relate the more abstract concepts introduced in the course to the details of the design and implementation environment.

Over the past year we have used student questionnaires to assess student response to the tools and attitudes towards design and implementation concepts. Students have also completed pre-test and post-test exercises, and a standardized series of assignments and exams. Each semester students are required to complete three design assignments prior to any programming. Nine programming assignments are completed, each emphasizing specific control and data structures. Each programming assignment must include a preliminary design document. While we have not completed our evaluation, we present a brief review of our findings to date.

We have seen an improvement in modular design. The number of designs submitted has increased and the quality and completeness of each design has improved. Students using the suite of tools show an increase in the efficiency of their programming as measured by time to complete programs. On the first few programs, time spent designing the solution has increased by 20 to 30 minutes, but the time spent programming has decreased on average by one hour. For larger programming assignments, students have again increased their design time by 20 to 30 minutes, but have decreased the time to complete a program by two to five hours. Design Tool facilitates the design process and provides the students with a framework to organize their problem solution prior to entering the actual coding phase.

Student use of the hypertext has increased overall utilization of the laboratory. While the time spent in programming may have decreased, the time working in lab and experimenting with programming concepts has increased. Students spend a slightly greater number of hours in lab each week, and the distribution of the time on various activities has changed. While less time is spent on actual assignments, there is an increase in time spent investigating the concepts discussed in class, as well as other concepts not yet introduced. It is not uncommon for 10% - 20% of the class to include design or programming concepts not yet introduced in class in their programs. This exploration seems to be facilitated by the hypertext with its easy access to example programs and conceptual information. Students have indicated that they find the increasing levels of detail in concept presentation, the multiple examples of each programming concept, and the executable examples supporting each concept useful. Not only can they view the text of the program, but they can copy the programs into the programming environment and observe the results of the execution.

Future Directions

To further support students in the mastery of design and implementation skills, we are developing an algorithm animator which will be integrated with the hypertext, Design Tool, and the Pascal programming environment. The animator's visual display of data and control flow will provide an additional representation of the concepts introduced in the course. As with the other components of the learning environment, the animator is designed to supply a bridge between abstract concepts and more concrete representations.

Acknowledgments

Intel Corporation donated equipment and support for the student laboratory. The graphic display program used in this work was developed under NASA grant NAG9-114 to the second author. Thanks to Iris Flores for writing and testing 210 programs and designs, and to Maybeth Shirah for help in converting Curricula '91 to hypertext.

References

Baile, F.K. (1991). Improving the modularization ability of novice programmers. *ACM SIGCSE Bulletin, 23*(1), 277-282.

Boehm, B.W. (1981). *Software Engineering Economics.* New York: McGraw-Hill.

Carrasquel, J., Roberts, J., & Pane, J. (1989). The Design Tree: A visual approach to top-down design and data flow. *ACM SIGCSE Bulletin, 21*(1), 17-21.

Hohmann, L., Guzdial, M., & Soloway, E. (1992). SODA: A computer-aided design environment for the doing and learning of software design. *Proceedings of the 4th International Conference, ICCAL '92: Computer Assisted Learning* (pp. 307-319). Berlin: Springer-Verlag.

Kiper, J., Lutz, M.J., & Etlinger, H.A. (1992). Undergraduate software engineering laboratories: A progress report from two universities. *ACM SIGCSE Bulletin, 24*(1), 57-62.

Koffman, E.P., Miller, P.L, & Wardle, C.E. (1984). Recommended curriculum for CS1: A report on the ACM curriculum task force for CS1. *Communications of the ACM, 27*(10) , 998-1001.

Mynatt, B.T., & Leventhal, L. M. (1990). An evaluation of a CASE-based approach to teaching undergraduate software engineering. *ACM SIGCSE Bulletin, 22*(1), 48-52.

Schweitzer, D., & Teel, S.C. (1989). AIDE: An automated tool for teaching design in an introductory programming course. *ACM SIGCSE Bulletin, 21*(1), 136-140.

Sidbury, J.R., Plishka, R., & Beidler, J. (1989). CASE and the undergraduate curriculum. *ACM SIGCSE Bulletin, 21*(1), 127-130.

Soloway, E. & Spohrer, J.C. (Eds.). (1989). *Studying the Novice Programmer.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Tucker, A.B. (Ed.). (1990). *Computing Curricula 1991: Report of the ACM/IEEE-CS Joint Curriculum Task Force.* New York: ACM Press.

Vessey, I., Jarvenpaa, S.I., & Tractinsky, N. (1992). Evaluation of vendor products: CASE tools as methodology companions. *Communications of the ACM, 35*(4), 90-105.

ERIC

The AT&T Teaching Theater at the University of Maryland at College Park

WALTER GILBERT
Computer Science Center
University of Maryland at College Park 20742-2411, USA

The University of Maryland at College Park, with the financial support of AT&T, has designed, constructed, and is using a classroom which features *technology to transform lectures and learning*. A network of computers, highly integrated with a broad array of audio/visual equipment, is set in an attractive atmosphere to encourage creative approaches to instruction. The room, *The AT&T Teaching Theater*, is an important step in extending to the lecture environment the benefits of technology that are revolutionizing other aspects of education.

The AT&T Teaching Theater stands out in many ways from the high-tech classrooms that exist elsewhere in higher education:

Goal: The major goal of the Teaching Theater is to improve the lecture process by changing it from its traditional unidirectional information flow to involve more collaboration. The availability of both anonymous and personalized intercommunication between the instructor and the students, and among the students themselves, creates unique and empowering classroom dynamics.

Availability: The AT&T Teaching Theater is a Campus resource; its use is not restricted to any particular discipline. Courses from Anthropology to Zoology are candidates for the kind of transformation the facility supports. Academic disciplines that have used it thus far include Anthropology, Computer Science, Engineering, English, History, Housing and Design, Information Systems Management, Library and Information Service, Mathematics, Psychology, and Zoology.

Support: An integral feature of the Teaching Theater is its support staff who are dedicated to keeping the facility current and responding to the creative requirements of faculty and students. With additional support from The Center for Teaching Excellence, another UMCP effort aimed at improving instruction, an active program has been established to assist faculty in preparing to use the Teaching Theater effectively. The room is never used without the presence of an experienced support person

Research: Support is available for designing, implementing, and evaluating research projects involving the AT&T Teaching Theater. Installed facilities provide multiple ways of recording class sessions for later analysis.

Background

The focus of the AT&T Teaching Theater on improving lectures is difficult for many people to comprehend initially. A common first reaction when presented with a classroom which has exceptional computing and audio/video capacity is to imagine an expanded computer classroom or dazzling multi-media presentations. It often takes an extended discussion to clarify the concept of really changing the way lectures are done. Also, since faculty have rarely ever experienced a lecture enhanced with lectureware, they can easily revert to focusing on the computers as computers instead of as lecture-assisting tools. The bottom line is that the AT&T Teaching Theater is intended to bring faculty and students closer together and to magnify the role of the instructor rather than substitute for the instructor and take over parts of the teaching process.

The lecture itself is probably the last of the educational processes to be substantially affected by technology. Every other area, from custom publishing of textbooks to doing homework, has been driven by, or at least improved by, technology. While the addition of sophisticated audio/video presentation and computer demonstrations has improved the quantity and diversity of information directed from the instructor to the student, these mechanisms still support only a one-way flow and offer little opportunity for students to benefit from the contributions of their peers. Additionally, they offer no support for the instructor to assess class understanding and the effectiveness and pertinence of his or her teaching efforts.

Available software permits an instructor to quickly assess the level of preparedness of the class by asking a question and receiving anonymous responses from all students, not just the traditional hand-raisers. These responses can be simple votes (yes/no), multiple choices, or short written statements. Software designed for group brainstorming is easily diverted to a variety of collaborative processes. Just the ability for all students to "talk" simultaneously and review their collective statements as the "discussion" proceeds can be a valuable process.

In the typical lecture setting, class response is often biased by the uneven distribution of students who actively participate in the classroom. This bias can color both the instructor's and other students' perceptions about the general level of understanding. Research at Maryland and elsewhere shows that classroom participation by women and minorities is consistently lower than average. One of the design goals of the Teaching Theater is to support and capitalize on simultaneous, anonymous input from students.

The AT&T Teaching Theater relegates sophisticated technology to a status similar to that of books and chalk, permitting it to be useful to faculty in virtually every discipline. The question a potential instructor must ask is: "Can my goals for this class be better accomplished using collaborative techniques and improved communication among the students, myself, and their classmates?" This is a much more difficult question to ponder than "Will my students better understand soil conservation by having access to the SoilCon software in the classroom?"

The AT&T Teaching Theater is a fertile field for developing and testing new teaching concepts and methodologies. It is a dynamic and evolving environment, and maintaining this evolution is one of the key features in its planning. The active support and development effort will keep the facility from backsliding to a nice computer classroom and will allow it to serve as an evolving prototype for other Campus efforts. Already, three additional classrooms based on the AT&T Teaching Theater experience are being designed for the new Business and Management Building and the new Plant Sciences Building. The lessons learned in the design and construction of the AT&T Teaching Theater are being used productively to reduce costs and to avoid subtle pitfalls in designing and equipping these new rooms.

ERIC

The AT&T Teaching Theater offers unique research opportunities in the study of classroom behavior, teaching strategies, comparative methodologies, effectiveness of technology, and other heretofore unaddressable issues. When appropriate, a detailed record of a class session can be made on video tape and most computer activity can be logged for future analysis. Support is available for faculty interested in using the unique aspects of the Teaching Theater for creative research.

Some Examples of Use

To schedule the room to teach a class for an entire semester, faculty must submit a proposal to the Steering Committee describing what they intend to do in the AT&T Teaching Theater that they could not do in a regular campus computer classroom, A/V room, or computer lab. Scheduling is managed by the project team.

Faculty have adapted to the AT&T Teaching Theater in a wide variety of ways, several never imagined during the design phase of the project. One professor has established a hypertext environment on all of the computers in the room. The home screen leads to the syllabus (which leads to detailed class notes, etc.), supplemental readings, a class roster including a picture and brief background of each student, a note-taking area, the list of assignments, several specific applications, and other items. He also integrates A/V sources into his presentations.

Another professor uses A/V for presentation of case studies and tools from VisionQuest, a group decision-support package, to support multiple discussion groups in the room. Her technique depends heavily on anonymous interactions.

Other professors use collaborative writing tools in creative ways, and one professor has developed a group software programming environment that permits students to collaboratively create and test complex programs.

Specifications

The AT&T Teaching Theater can accommodate 40 students at 20 custom-designed student desks. The desks are cantilevered so that there are no supports to restrict chair movement; the front row of desks is wheelchair accessible. The computer monitors are recessed into the desk tops so that they provide no visual obstruction. Student chairs are five-caster swivel chairs which promote student interaction and somewhat compensate for the disadvantages of the theater style of room layout.

The instructor's console is designed to encourage experimentation with the layout of equipment and controls.

The room is 39'x24', this geometry restricted the configuration of student desks to four rows of five. Each of the last three rows is tiered five inches above the row in front of it to improve sight lines to the front. The ceiling is low: 7'6" in the rear and 9' in the front.

The AT&T Teaching Theater is in room 3140 of the Engineering Classroom Building, space generously provided by the Electrical Engineering Department. There are two adjacent support rooms. One houses all the computers and the other houses the A/V equipment and provides space for the support personnel. The monitors, keyboards, and mice are operated over cables up to 45 feet long. Removing the computers from the room has many, many advantages.

Computers

Each of the 20 student desks is supported by an AT&T 25MHz 386-based computer with math coprocessor, 8MB of memory, and 142MB of local hard disk space. Each computer has a 17" high-resolution color monitor, keyboard, and mouse. Windows 3.1 is the normal operating environment. The instructor has two computers, each 33MHz 386-based computers with larger memory and disks. The network server is a 33MHz 386 with 32MB of memory and about 3GB of disk space.

Networks

All the computers in the AT&T Teaching Theater are attached to the Campus optical fiber-based (Internet) network and to an AT&T StarLAN network with a local Novell server. These connections give students and faculty access to all network-based information sources and collaborative software applications.

Students and faculty can access their file storage on the Novell server from any of the approximately 40 computer rooms on Campus which have Novell networks, particularly those in the many workstation labs. In this way, students can print materials acquired during class sessions without having to wait in line after class. They can also transfer files to or from their Teaching Theater disk space, thus permitting them to set up class materials without having to manipulate floppy disks during class. The availability of FTP permits dial-up access to files.

Audio/Video Projection

All the video equipment in the Teaching Theater, even the "overhead" projector, produces NTSC (standard television) images. All images can be projected on either or both of two 4½'x6' rear-projection screens in the front of the room.

The image from any computer in the room can be projected onto either or both screens; the two projectors support up to 1024x768 resolution. The instructor can switch the image from any student monitor to the front desk and can switch the image from the instructor's monitor to any or all student systems. The students' keyboards and mice can similarly be switched. The instructor can also blank all students' screens.

All equipment is controlled by a convenient touch screen on the instructor's desk or a hand-held remote control. The audio/video equipment in the AT&T Teaching Theater includes:

two high-resolution projectors	an S-VHS video tape player
a U-Matic (3/4") video tape player	a laser disc player
a broadcast television antenna	connection to the campus video cable
a 35mm slide projector (video image)	an overhead projector (video image)
a compact disc (CD) player	an audio tape player
a stereo speaker system	a closed captioning decoder

Recording and Teleconferencing

The Teaching Theater has three remotely controlled video cameras, one in the front of the room and two in the rear. One of the rear cameras is normally focused on the whiteboard, the other two can be used to monitor classroom activity. All three cameras can be used for recording and teleconferencing. Several ceiling microphones pick up student voices; the instructor can wear a portable, radio-linked microphone. Acoustic feedback is not possible because the microphones are not linked to the speakers in the room.

The cameras and microphones are controlled from a production console containing an S-VHS recorder, special-effects generator, and related equipment. The console is in the rear-projection area.

Environment

By removing the computers and audio/visual equipment from the classroom to adjacent areas, the classroom was optimized for human comfort while the equipment areas were optimized for electronics. Specifically, the furniture does not have to accommodate computers with their noise and cooling requirements, the acoustics do not have to provide

Room lighting is divided into eight separately controllable zones: six incandescent, each continuously variable from full off to full on, and two fluorescent, on/off. An infinite variety of lighting moods can be created and controlled by the touch-screen controller on the instructor's desk.

The acoustics of the AT&T Teaching Theater have been carefully optimized for speech. In addition, the room is very well isolated from outside noise and has a custom-designed air handling system to minimize internal noise. This includes lined air ducts, turning vanes, and a low air velocity.

All computer and key A/V equipment is powered by a 12.5KVA uninterruptable power supply (UPS). This isolates them from transient power interruptions and damaging spikes. It also provides a comfortable shut-down period when long outages occur.

Future Plans

The Campus is currently constructing the IBM Teaching Theater as part of a Total Quality grant from IBM. It very closely resembles the AT&T Teaching Theater in design, function, and support. Its seating is in two concentric horseshoes instead of four theater-style rows. This configuration is preferred because it promotes collaboration and interaction between students. The geometry of the AT&T room did not permit the use of a horseshoe configuration.

Construction is beginning on a new Plant Sciences Building which will contain two similar classrooms. The larger will closely resemble the AT&T and IBM Teaching Theaters while the smaller is a seminar room seating 12 in a single horseshoe. They share a common A/V-computer room. They will available in Fall, 1995.

Layering Tools for Analyzing Video Data:
Making Sense of Inscrutable Video Using a *Significance* Measure

RICKI GOLDMAN-SEGALL
Faculty of Education, University of British Columbia
Vancouver, BC, Canada V6T 1Z4

This paper proposes a new orientation in the development of software tools for researchers who code, annotate, and analyze video data. The direction presented addresses a core problem in the use of video data: how to make sense of the chunked clusters of video that are difficult to interpret contextually and even more difficult to interpret when sharing data with collaborators. The claim made is that dealing with video data is fundamentally different in kind than working with text-driven data due to the multi-grained nature of video. As researchers code and annotate their data, video chunks can be delineated and weighted. These layered structures will help us "see" what video data are significant, and for whom, to the total body of video data. Significance is defined by the users who will decide for themselves upon categories that relate to the task at hand. Adding the significant measure will aid the work of collaborators sharing the same video and text by building layered interpretations.

The Problem:
Analyzing the Significance and Contextual Nature of Video Chunks of Data is Complex

A theoretical approach for understanding the significance of specific chunks of video data to the entire video study needs to be developed.[1] This situation is an impediment to our relying upon video data to draw conclusions. The problem I would like to address is that, until we find a way to determine the significance of certain video chunks to a large video stream, the conclusions we draw may not be representative of the whole body of work. To overcome this problem, we need first to examine why working with video is inherently different from working with text data, then to explore the needs of users who are working with video data, and lastly to propose a model for designing software with visual aids in a database that will help determine the significance of a given chunk of data to overall objectives in a given study. What is needed are tools that help us decide whether a closing of the eyelids is a twitch, a wink, or a conspiratorial communication, as Clifford Geertz might ask (Geertz, 1973). Geertz appropriated the term *thick descriptions* (first coined by Oxford scholar, Gilbert Ryle [Ryle, 1971]) to address issues of validity in ethnographic descriptions of cultures. For Geertz, a thick description of the closing of the eye lids would enable the reader of a text to come understand what that gesture might have meant for the blinker. However, in *Works and Lives*, Geertz underscores that the meaning is tied to the writer's version of the event (1988). In dealing with video data, thus, the meaning is ascribed by the writer of the text. To build tools for video analysis, we must come to terms with who is the video researcher, who shot the video, when, where, and for what purpose—paying particular attention to the context in which the event was recorded.

[1] In Goldman-Segall (1989b, & 1990) a theoretical model of building layers of video data to thickly describe the meaning of videotaped events was first recommended. These works describe the refining of a video analysis tool, *Learning Constellations 1 (LC 1)* which was originally designed in 1989 by my colleagues and I at the MIT Media Lab to analyze specific video data I had collected of children in a Boston elementary school. *LC 2*, is a generic tool for video analysis, is being built in MERLin, the multimedia ethnographic research lab in the Faculty of Education at UBC. The focus of MERLin is to develop tools for an emerging community of video ethnographers.

Conventional text-based computational tools will not handle video data because video data are inherently different from text data. Compare the five minute text transcript of an interview of a child named John with the data from a five minute interview of him recorded on video. The text transcript is relatively easy to code, annotate with comments, file, store, retrieve. To analyze the data contextually, the researcher who is trying to come up with a theory of a child's use of the word "amazing" might have to consider the following: How is this word used to link ideas? Does saying "amazing" give John time to pause in order to think about the next topic? How does this word show curiosity about the world? What do we know about John's interest in one subject compared to another? We can easily imagine the researcher (who is analyzing only text) searching through hours of text transcripts to check for other instances of the use of '"amazing." It would not be too difficult to compare John's use of the word with another child's. It would not be problematic to search and link relevant annotations of other researchers who may have commented on John's use of "amazing." Making textual annotation to a chunk of data would not necessarily present a problem.

To analyze this video stream, we need to be able to interpret not only what John is saying to us when he uses the word "amazing," but what he is doing — hoping that what he does will give us clues about what he means. We need to be able to determine which pieces are significant, if we are to make meaning for video later. Therefore, what we need are tools that will "weigh" or "layer" the significance of video data so that our conclusions are based upon what a child means when he says "amazing" and then slaps his hand on his thigh and pauses, staring into space. To share our interpretations of these kinds of events, we need semantic visual tools for communicating to others what we thought was significant about what we found.

Describing the Obstacle:

Interpreting Visual Images is More Difficult than Interpreting Text

What needs to happen in the development of software tools so that we will be able to interpret video as intuitively as we do text? To answer this question, we need first to overcome a central obstacle: interpreting visual images is fundamentally more difficult than interpreting text.

The problem is that we have not found a systematic method for interpreting visual images. Landau (1989) in *About Faces: The Evolution of the Human Face* explores the evolution of facial gestures of humans and primates. In this book, one finds a multitude of illustrations of facial expressions and a meaning assigned to each. As is clear from the title, Landau believes that these gestures are the result of our earlier history, linking us with primates. (One example is the facial gesture with the lips exposing the teeth. According to Landau, this "smile" was once the gesture [illegible] potential enemies to stay away.) Landau's interpretations of facial gestures do not address [illegible] and cultural differences. The issue facing a researcher using visual data is to find a way to [illegible]erpret a given gesture or visual representation. If we are to make meaning from our visual data, we need less to have a universal taxonomy of what gestures mean, and more a method of exploring the meaning for that person within a given culture.

What does it mean to interpret visual images and attach a meaning to what we see? The job of interpreting signs is as old as our humanity. The earliest records we have that people made sense of the world through interpreting images are the paintings on cave walls. Some paintings told stories about great hunts; some taught others where to look for food or shelter; and some communicated to other travelers the best route to another settlement. In this last example, these early paintings were navigational tools for exploring the physical and spiritual terrain. In fact, what we do today with multimedia software tools that use visual images is quite similar. We construct narratives; we educate; and we try to design pathways for navigation through data.

Our historical obsession with the power of the visual representation is perhaps most clearly described in the bible: one of the ten commandments is not to create or worship a graven image. In pre-biblical times, each god or goddess had a physical manifestation. In becoming a monotheistic religion, Judaism and then Christianity forbade the worship of the representation of deities or even the representation of one deity. Elliot Soloway, when talking about children using multimedia, says that "those who create the images are empowered" (Soloway, 1992). The fear of the high priests was that people should not have the power to create their own representations.

The legacy of the biblical non-pictorial texts is that they lessened the multiplicity of interpretation that was intrinsically connected to the power of visual icons. However, it should be pointed out that the Byblos alphabet of the ancient Semites is a system requiring a great deal of interpretation — it is a system of letters standing for sounds that have no letters for vowels. Illich and Sanders (1989, p. 12) say that meaning is "breathed into" the semantic Byblos text by the reader. This is reminiscent of Jerome Bruner's (1986, p. 7.) statement about that the text is "loosened in the mind of the reader." In the Byblos text, meaning is understood by analyzing the word in relation the whole sentence and chapter.

When Greek merchants took this writing system back to Greece, they altered the script by adding letters for vowels.

> The Greeks froze the flow of speech itself on the page. The [Hebrew] scroll had been sounded thus far through an act of interpretation of the letters... The Jew searches with his eyes for inaudible roots in order to flesh them out with his breath. The Greek picks the sound form the page and searches for the invisible ideas in the sounds the letters command him to make (pp. 12-13).

Historically, representing knowledge ~~moved from~~ reading pictures (such as, Egyptian hieroglyphics and Mesopotamian ideograms) to the Semitic system of reading words with consonants that needed to be contextually read to be understood. When the Greeks assigned a fixed vowel to each word, the role of the reader became less interpretive. With this progression, meaning was less likely to be constructed by the reader and more likely to be assigned by the author of the text. The challenge of the re-introducing the visual into our texts is that interpretation may once again become the prerogative of the viewer.

Addressing the Problem:
Building Tools to Organize and Interpret Visual Data

Hierarchical, Narrative, Relational and Hermeneutical Approaches

The power to build images and manipulate visual data has changed hands with the use of the computer. The computer has become a tool to manipulate visual or text-based items; it is "a facility for machine support of arbitrary cross-linking between items." (Conklin, 1987; Drexler, 1987). Those items or objects can be text, sound, or visual units. A range of approaches has been taken to deal with the manipulation of video and text data. I will describe several approaches that are hierarchical, narrative, and relational in structure.

A hierarchical approach is exemplified in a multimedia authoring system called *HyperG* (Maurer, 1992). Maurer and his team at the University of Technology in Graz, Austria are concerned less with building powerful meta-systems for accessing a large amount and an extensive range of data. In *HyperG* each item — a word or a picture — is attached to a structure. A structure belongs to a parent structure. Navigation is handled either by following the hyperlinks attached on a document or by browsing through what Maurer calls "the structure hierarchy" consisting of *documents*, *clusters*, and *paths*. The design of *HyperG* is an engineering problem: how does a given structure support various activities? Maurer tackles the problem by establishing a hierarchical model for searching and browsing data of all kinds.

In a more narrative vein, Davenport, Smith, and Pincever (1991, p. 69) define linking as a "user-directed form of storytelling." Davenport and colleagues have approached the problem of how to make stories of digital video by applying the cinematographer's point of view and by building tools that a cinematographer would need to build film narratives. They describe what happens to the film editor when putting together individual shots to tell a story. Then, they ask what would happen if one could build "machine-generated, multi-threaded parallel narratives that users can adjust through linking" (1991, p. 73). Would multi-threaded parallel narratives build layers or strata upon the base story-line, they ask? This notion of layering the multiple meanings of a film event or stratifying the content was first described by Goldman-Segall (1989b), while working with Davenport. She built upon Geertz's and Ryle's use of the term, *thick descriptions.*

> We hypermedia ethnographers may want to find ways to build systems which delineate the consistency of the smallest unit, or as it could be called, the *content granularity*. . . Geertz's *thick descriptions* may be the conceptual tool we need in our designing of interactive systems. *Thick descriptions* are descriptions which are layered and textured enough for readers/viewers to draw conclusions and uncover the intentions of a given act, event, or process. (p. 118)

Other approaches suggest more relational and interactive methods of constructing meaning from visual data. Lippman (1989), who first coined the word, *granularity*, lists five conditions of an interactive system. These conditions are: interruptibility—ability to interrupt and maintain focus; *granularity*—ability to define the smallest grain or chunk; *limited look-ahead*—ability to see the next step; *graceful degradation* —ability to have a graceful way of concluding; and at minimum, the *illusion of infinitude* in choices when describing interactive linking tools. The underlying assumption in the interactive approach is that the structures should to respond like neural networks. In this paradigm, what we know is constructed by *interacting* with and relating to visual and text driven media in diverse ways.

Building tools to organize and link visual data is quite a different task from building to interpret them. Interpretation tools require more than adopting a structural meta-approach, whether it be hierarchical, narrative, or relational. What I propose is that tools be designed with a hermenutical approach where the goal is to make meaning by understanding the significance of the bits and pieces. The reader/viewer co-constructs meaning by assigning a weight to each piece. This approach ties Lippman's notion of granularity to Geertz's (1973) notion of *thick descriptions*. You get closer to understanding the richness of an event by first being able to describe it grain by grain. The contextual additions to the grain or atom add the texture of the meaning. In fact, the exercise of assigning meaning is similar to constructing the meaning of a poem. The reader first "finds" the literal sense of the events by tracing the storyline. (These elements are the grains or chunks of data.) To appreciate the meaning, the reader layers her or his understanding by "breathing into" (as Illich and Sanders would say) the tone, the subtleties, and the genre.

Another 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 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. Users may build analyses that "fall into the same range" (Geertz, 1973) but they are not necessarily the same. For example, different users may agree that John's use of the word 'amazing' represents awe and wonder, but they may still think other interpretations are possible. New applications of video tools will need to be developed to help users incorporate the plethora of varied and various interpretations.

Systems for Annotating Video Data: *CVideo; VANNA: Learning Constellations*

Relevant work in the development of tools for chunking, linking, annotating, and navigation through video data is not very extensive to date. Harrison and Baecker (1992) divide the tools into 1) those which are notation systems for representing material from videotapes— such as the Heath Notation System (Heath, 1986; Heath and Luff, 1991), and 2) those which are video analysis systems for controlling the video, coding it in some way, and then having some additional tools to help in the analysis. This second group includes: GALATEA, a graphic system for analyzing and marking up video of biological phenomenon (Potel, Sayre, and MacKay, 1980), The GroupAnalyser that allows those interested in analyzing group dynamics the ability to create and display animated diagrams (Losada and Markovitz, 1990), EVA for both on-line real time coding and off-time more detailed coding (Mackay, 1989), U-Test designed as a usability testing system for real time on-line coding (Kennedy, 1989), and Virtual VCR with a system of tagging and making short comments (Buxton and Moran, 1990). This second group of video analysis tools also includes three other systems:

CVideo

Randy Trigg from Xerox Parc and Jeremy Roschelle from the Institute for Research in Learning (IRL) in Palo Alto have been designing an annotation system since 1989 (Trigg, 1989; Roschelle, Trig and Pea, 1990). *CVideo*, previously called *Videonoter*, was prototyped by Roschelle and released as a commercial package in 1992. With *CVideo*, a researcher can keep a running log of video by marking in and out (start and end) points of video and being able to locate previously logged video easily. *CVideo* also enables researchers to add both transcripts and annotations in a window. One can easily find a location on the video by using a moveable bar to scan the video. Unfortunately, *CVideo* does not link chunks of video together into groupings. Moreover, it does not include a specific feature of *VideoNoter* which allowed the user the ability to move categories in what looked like a miniature database. However, *CVideo* keeps a clear linear record of video data for further analysis at a later date. Another advantage to *CVideo* is that it can access and control video from Hi8/VHS video sources and videodisc players, a feature to also be found in the next two applications as well.

VANNA

VANNA is the Video ANNotation and Analysis Tool (1991-1992) prototyped and designed by Harrison working with Baecker at the University of Toronto. Not available yet as a commercial package, it is a tool to code and annotate video. In its most portable configuration, it can be easily used hooking up a video camcorder with an external mini-monitor to a Mac Powerbook using the playback on the video camera. Harrison has built in an abundance of input devices ranging from the finger on a touch screen, the mouse, the keyboard, to a digital stylus. Its most interesting feature is that it supports real time annotation and coding of data. It also has a system of annotations that can be added later. *VANNA* was designed following a series of human user tests, including brainstorming sessions, with many versions occurring over a short time.

Learning Constellations 1:

Learning Constellations 1 (1989a, Goldmann-Segall et.al.). *LC 1* was the first video annotation system built which supported the analysis of an entire body of data of a researcher using ethnographic style video data on videodiscs. Several hours of selected video data on videodiscs were analyzed in order to build detailed case studies. What made these case studies unique was that a gesture or subtle movement could be viewed as the researcher was writing about the children Views changed upon reviewing the same data. However, when the researcher needed not only her mental passive image of John to guide my description, she could, in an instance, have the raw video in front of her, reflecting back the initial experience.

In short, using the first version of *Learning Constellations*, the researcher was able to: 1) choose the start and end points called the chunks and choose the ordering of chunks, 2) search through transcripts, topics and participant names attached to each video chunk, 3) link video together into clusters called *constellations*, 4) browse, view, and/or build annotations and groupings of video, 5) analyze the data from many diverse points of view, 6) examine the same documentation with collaborators adding new levels of meaning by including their own written observations to the existing descriptions, and 7) use the tool as a presentation device to show selected video data.

Revisions to *Learning Constellations 1*

Revising *Learning Constellations* required more than the idea of what needed improvement. It required a laboratory environment and a collaborative group of video ethnographers to direct the design. This new lab, MERLin, was established in 1991. Working with Goldman-Segall, Monika Marcovici, programmer, describes the new design of LC2 in the following way:

> The overall goal in the redesign was to give more power to its users, not only by adding new features to it, but by making existing functions easier to access and more flexible. The advantage of an object oriented environment such as HyperCard, the development platform for *Learning Constellations*, is the possibility to free the user from the types of constraints such as orders of operations, and remembering specific commands, to permit him or her to create, select and manipulate data comfortably. Thus, the revision of *Learning Constellations 1* exploits the modularity of HyperCard more fully, in order to fulfill the goals of the new design. Continuing this with a database, File Flex, the database used has extended the usual capacities of Hypercard.
>
> **SPECIFIC CHANGES**
>
> **Generic Nature** The previous version of *Learning Constellations* was not easily applicable to other areas of research. The data were inextricably intertwined with the actual program. The redesign of *Learning Constellations* separates the program from the actual data, making it possible to create new research environments for other applications.
>
> **Data Structure** The primary redesign involved a change in the data structure. A relational database engine is linked with the program and operates in the background. This results in significantly faster data searches permitting more sophisticated types of queries, links, and ways of sorting the "chunks" of data in the database. In the previous version, the data was first classified by its type (video or text), and then tracked as an entry in a list. In the revised version, a star can be any discrete element of data (e.g., video, text, sounds, pictures, or digitized video sequences) and can be described, found in a search, browsed, sorted, edited, deleted, and annotated. Regardless of the types of information involved, each star chunk is accessed and played in an identical manner; data are analyzed in terms of their content and meaning, regardless of type or the way that it was recorded.
>
> **"The Significance Measure"** A critical goal of *Leaning Constellations* is to assist in the navigation through countless pieces if data, to draw out significant pieces and to connect them in order to help draw conclusions and build theories. The "Significance Measure" is a tool which researchers may use to rate the significance of a certain topic or participant in a chunk of information. The significance is recorded by sliding a button on a colored vertical scale, from a saturated color at the very top (corresponding to a rating of "10") to an unsaturated color at the bottom. The color matches the color that has been assigned to the category. In this way, the researcher is provided with visual clues in her or his building of significance. Searching for data containing a certain keyword will not only provide a random list of matching data, but will sort the data in a list according to its rating in relation to the criteria in question. In addition, information such as "the most important themes" for a specific contributor, or author" will now be possible (See Illustration #1).
>
> **Pop Up and Pull Down Menus** One possible source of error and frustration in *LC 1* was that a majority of the data in the system was entered by its users typing into it. Possible errors could occur anywhere from the typing of the login name, in-points and out-points of a video selection (usually a five digit number), to the typing in of the participants and the names of topics. Any of these errors could cause data to be lost during searches where the criteria are spelled differently. By creating "point and click" pop-up and pull-down menus whenever possible, there is substantially less typing on the user's part, and the chance for error is minimized. Selecting participants and topics from a menu rather than typing them is also less straining for users and encourages more accurate indexing of the data.
>
> **Annotation with sound or text.** In keeping with the consistency in use of the various media available, since text and video notes are presently available as annotation tools in *LC 1*, sound annotation is also being added to the new version.
>
> **Text Format** This feature will permit simple import of existing word processed documents into the Constellations database.
>
> **Customized Search** As in the previous version, it is possible to navigate through the star chunks by clicking on a given participant or topic from the list or by going to the next star with this information. The new system accommodates multiple search criteria, such as searching for a group of stars,

constellations, or annotations created by certain authors, referring to a number of participants and topics. The researcher simply selects the desired topics, participants, or authors from their index columns, and a list of stars matching the highlighted specifications are available for browsing and navigating through the data. Moreover, the clicking changes the data in the quadrant but the page does not keep changing.

LC 2, an exemplar of this is already been implemented. The model works in the following manner: A user navigates through a chunk of video creating or browsing through individual chunks, but not leaving a base "home card" with four main Quadrants (See Illustration #1).

In the upper left quadrant, the user: 1)selects the data type (text, video, picture sound); 2) chunks out the data according to its type by entering start and end points; 3) describes each chunk by filling in topics, participates and transcripts, and; 4) rates the significance of each topic and participant, according to her or his own particular research agenda. In the upper right Quadrant, the data always appear. Whether a Quick Time movie or a pict file, the data appear in the same place. In the lower right Quadrant, the database is found. A complete list of users, topics and participants remains visible. Scrolling windows and subheadings make the visual search possible. Clicking on items from any of the three categories in the database brings up a list in the lower left Quadrant. Therefore, the lower left Quadrant is the table with those star chunks that were found in the search conducted in the lower right side. Moreover, when clicking on any star chunk in this Quadrant, the user is brought to that chunk. The data appear in the upper right Quadrant. The local knowledge information fills into the upper left Quadrant.

In our next version, labels of stars will "live" in a three dimensional co-ordinate system (resembling the galaxy under investigation) where each axis represents another measure. The researcher will be able to create a view of the visual data that expresses the relevance of each section of video. In a possible scenario, the three dimensional graph could act much like MultiLogo agents — with individual stars reacting to each other in much the same way as do Resnick's (1991a, 1991b) concurrent turtles do.

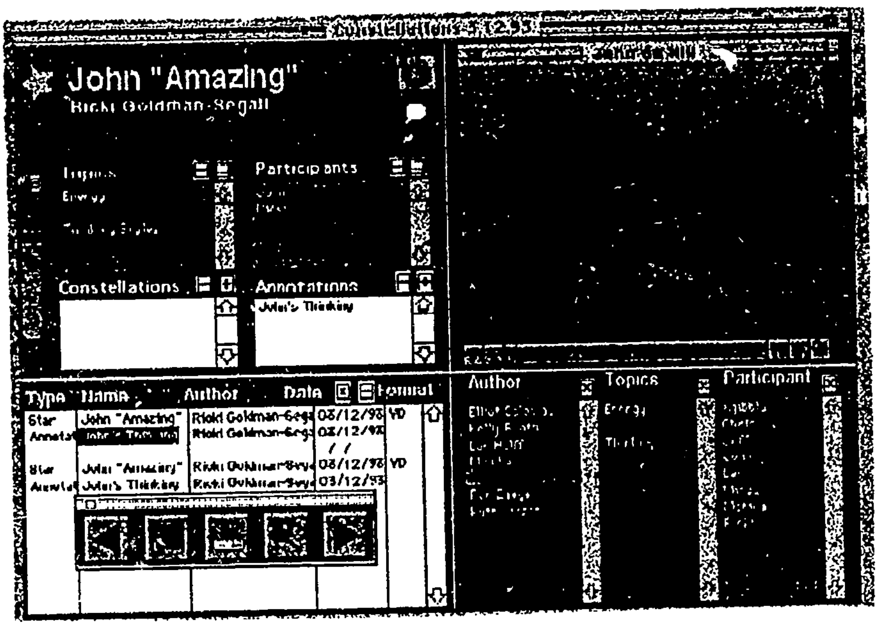


Illustration #1: *Significance Measure Icon* is the vertical bar in upper left Quadrant (directly below Star Icon). It rates the importance of items in the Topics, Participants, Constellations, and Annotations Windows. Data is displayed in upper right Quad. Database in lower right Quad. Results of searching database are displayed in lower left Quad. Clicking on an item displays item and brings up specific data.

BEST COPY AVAILABLE

Projections:
Significance Tools that Layer Data

In *Learning Constellations 2*, laying tools have been designed so that they visually represent the significance of specific video data to the body of the data. As a result, layered frameworks for how to build systems that can "weigh" specific data are being developed.

In the future, those of us concerned with video data will need to pay attention to the problems that have already been solved using existing tools and methods of video analysis (e.g., *CVideo*, *VANNA*, *Learning Constellations*, and more recently, Elliot's [1992] *video streamer* and *shot parser*). Moreover, we will need to design and implement a mechanism that communicates from the annotations to a visually-based database structure where layers of data can be easily accessed, simultaneously taking the user back to the contextual information (which supports a general understanding of the meaning of the video event). In other words, we need to begin to address how we analyze video streams of data after we have placed them, accessed them, and linked them. In the testing phase, we will need to study the ways in which individual researchers from a variety of domains learn to know which aspects of their video data are significant to them. Then, we need to implement a system of layered descriptions that would provide them with visual clues.

Layering the significance will address the needs of either the individual researcher or multiple users sharing video. A team of users can, at a glance, check out how important a given chunk of video or text is to a collaborating colleague.

Significant tools will substantially support the work of multiple users interested in building more sophisticated methods of managing the multi-grained intangible nature of this electronic medium.

This research is supported by a strategic research grant (#5-81457) from the Natural Science and Engineering Research Council (NSERC) of Canada.

Acknowledgments:

I would like to thank the graduate students and research staff who work in MERLin (Multimedia Ethnographic Research Lab) for making this paper possible. Monika Marcovici, Scott Flinn, Ajibola Ajayi, Lar Halff, and Bruce Finlayson are a team that has shown me how important collaboration is in the designing of any "significant" significance tool. The doctoral students who are putting their faith in *LC 2* are also a serious part of this endeavor — Akosua Addo, Deborah Begoray, and Susan Taylor Alling to name a few. I would also like to thank Dr. Kellogg Booth of the Computer Science Department for his continual support.

References

Buxton. B. & Moran, (1990). EuroPARC's integrated interactive intermedia facility (iiif): Early Experiences. In *Proceedings of the IFIP*. WG8.4 Conference on Multi-User Interfaces and Applications.

Bruner, J. (1986). *Actual Minds, Possible Worlds*. Cambridge, MA & London, England: Harvard University Press.

Conklin, J. (1987, February). A survey of hypertext. *MCC Technical Report* No. STP-356-86.

Davenport, G., Smith, Thomas A., and Natalio Pincever. (1991, July). Cinematic primitives for multimedia. *IEEE Computer Graphics and Applications*.

Drexler, K. (1987, December). Hypertext publishing and the evolution of knowledge. Palo Alto, CA: Stanford, Computer Science Dept.

Elliot, E. (1992). Multiple views of digital video. Unpublished paper. Cambridge, MA: MIT Media Lab.

Geertz, C. (1973). *The Interpretation of Cultures*, New York: Basic Books.

Geertz, C. (1988). *Works and Lives: Anthropologist as Author*. Palo Alto, CA: Stanford University Press.

Geertz, C. (1992). Local knowledge and its limits: Some obiter data. *The Yale Journal of Criticism* 5 (2).

Goldman-Segall, R. and V. Orni Mester and D. Greschler. (1989a). *Learning Constellations* [Ethnographic Software and six videodiscs].

Goldman-Segall, R. (1989b). Thick Descriptions: A tool for designing ethnographic interactive videodiscs. *SIGCHI Bulletin* 21 (2).

Goldman-Segall, R. (1990). *Learning Constellations*: A multimedia ethnographic research environment using video technology to explore children's thinking. Ph.D. dissertation, Massachusetts Institute of Technology.

Goldman-Segall, R. (1991). A multimedia research tool for ethnographic investigation. In I. Harel & S. Papert (Eds.), *Constructionism*, Norwood, New Jersey: Ablex Publishers.

Harrison, B. L. & R. M. Baecker. (1991, May). Designing video annotation and analysis systems. In *Proceedings of CHI'91*. ACM, New York.

Heath, C. (1986). Body Movement and Speech in Medicine Interaction. Cambridge, England: Cambridge University Press.

Heath, C. & P. Luff. (1991). Disembodied conduct: Communications through video in a multi-media office environment. *Proceedings of CHI'91*. ACM, New York.

Illich, I. and B. Sanders. (1989). *ABC, The Alphabetization of the Popular Mind*. New York: Vintage Books.

Kennedy, S. (1989). Using Video in the BNR Utility Lab. *SIGCHI Bulletin* 21 (2).

Landau, T. (1989). *About Faces: The Evolution of the Human Face*. New York, London, Toronto, Sydney and Auckland: Anchor Books.

Lippman, A. (March, 1989). Lippman on interactively. *matures*. Special addition on Multimedia.

Losada, M. & S. Markovitz. (1990). GroupAnalyzer: A system for dynamic analysis of group interaction. *Proceedings of the 23rd Annual Hawaii International Conference on System Sciences*, IEEE Computer Society.

Mackay, W. (1989). EVA: An experimental video annotator for symbolic analysis of video data. *SIGCHI Bulletin* 21 (2).

Maurer, H. & I. Tomek. (June, 1992). Helping the user to select a link. In *The Institute for Information Processing Series*. University of Gratz. Report #339.

Potel, M.J., Sayre, R.E., & S.A. MacKay. (1980). Graphics input tools for interactive motion analysis. *Computer Graphics and Image Processing* 14.

Resnick, M. (1991a). Animal simulations with *Logo: Massive parallelism for the masses. In J. A. Meyer & S. Wilson (Eds.), *From Animals to Animats*. Cambridge, MA: MIT Press.

Resnick, M. (1991b). MultiLogo: A Study of Children and Concurrent Thinking. In I. Harel & S. Papert (Eds.), *Constructionism*, Norwood, New Jersey: Ablex Publishers.

Roschelle, J., Pea, R., & R. Trigg. (March, 1990). VideoNoter: A tool for exploratory video analysis. *IRL Technical Report* No IRL90-0021.

Ryle, G. (1971). *Collected Papers*. New York: Barnes and Noble.

Soloway, E. (1992). Reading and writing in the 21st Century Presentation at ECCO'92, Toronto, Canada.

Trigg, R. (October, 1989). Computer Support for Transcribing Recorded Activity. *SIGCHI Bulletin* 21 (2).

The Knowledge Board: Using Hypertext as an "Intelligent" Workspace for Writing Issues-Based Prose

GEORGE GONZALEZ
Galaxy Scientific Corporation
Brooks AFB, TX 78235-5352 USA
PATRICIA A. CARLSON
U.S. Air Force Armstrong Laboratory
Brooks AFB, TX 78235- 5352 USA

This paper describes the conceptual design and prototype implementation of a set of "knowledge templates" which integrate a body of texts (whose knowledge footprint and interconnections have been made explicit through hypertext) and a set of cognitive-support tools for composing an issues-oriented, analytical text which either codifies, extends, or mediates among various source texts. Because this "intelligent" workspace serves as an engine, it can provide scaffolding for many design or problem-solving tasks characterized by these cognitive activities:

- Verbal reasoning skills, such as comparison, contrast, analysis, synthesis, and identifying patterns in bodies of information.
- Metacognitive abilities, such as value assessment, discrimination, classification, judgment, as applied to qualitative and quantitative decision making.

System Overview

The Task: The rhetoric task modeled by this software involves a type of writing familiar to managers, investigative reporters, engineers, and researchers -- to name just a few. In issue-based writing, source documents become the raw materials in composing a position paper, an evaluative summary, an interpretive response. The **Knowledge Board** is the prototype of a comprehensive, integrated writing environment providing cognitive support for professionals who create complex documents as part of their job. It differs from many other computer-aided writing tools in that it is an end-to-end development tool. It assists the writer throughout the process, from generation of ideas to production of connected prose.

Cognitive Models: Reasoning with language requires not only ingesting information but also understanding content well enough to draw inferences and to apply concepts in different contexts. These critical skills, augmented by domain-specific knowledge, are the foundation of mature learning strategies and expert behavior patterns for reading and writing. Several major cognitive studies in the past decade have examined how verbal reasoning happens (Anderson, 1983; Bereiter and Scardamalia, 1987; Flower and Hayes, 1980). This project uses a problem-solving paradigm to explain writing and emphasizes the power of representation or reification in helping the writer to move gracefully through the activity spaces of the task.

Software Components: As indicated by Figure 1, the complete **Knowledge Board** uses Bereiter & Scardamalia's (1987) notion of a *content space* (essentially, summarizing, analyzing, and synthesizing information about the topic) and a *rhetoric space* (essentially, planning and organizing the domain information into a logically and stylistically appropriate formal artifact). Six distinct cognition enhancers work in tandem to help with the mental overload. Each element is more fully explained in the next six segments. The artificial neural network technology which preprocesses the textbase into a cluster hierarchy is described in Carlson and The (1991).

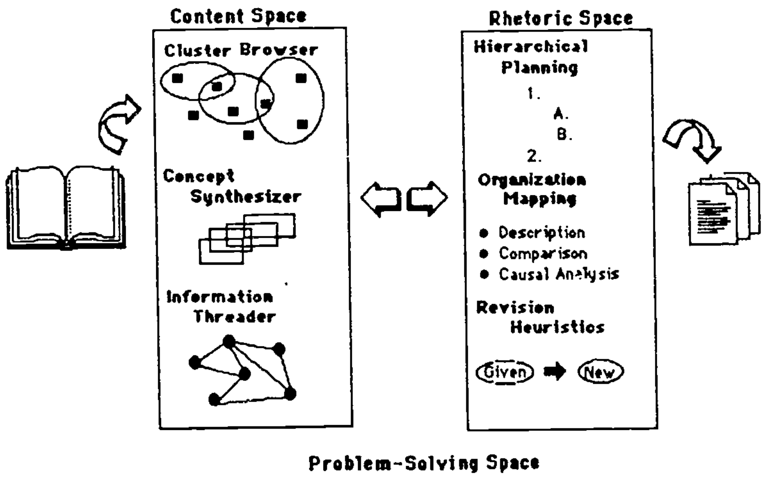


Figure 1. System overview of the Knowledge Board

Cluster Browser

Analogous to the convention of "view" in DBMS, this component helps the writer to locate specific concepts in a collection of source texts. (The assumption is that the writer has given the source documents a cursory reading prior to beginning a writing task.) Unlike traditional print technology, the **Cluster Browser** guides navigation through a textbase by extracting a knowledge footprint for the entire corpus of source materials.

Each source text is processed through an artificial neural network trained to categorize sentences based on their level of specificity. The resultant database represents a deconstruction of the source texts: linear sequence is abandoned and boundaries between documents are dissolved. Sentences are stored as collections of statements based on their level of specificity (Level 1 . . . n).

The writer queries the resultant database using keywords and phrases. This produces a visual representation of the "hits" -- using color coding, positioning, and labeling to indicate source text and potential content for each targeted statement. In other words, the user can determine from this high-level abstraction all probable topic sentences containing a keyword; all probable sub-topic sentences containing a keyword; all probable support or example statements containing a keyword; and from which source text each statement comes. At the resultant concept browser map, the user can click on any button and be taken directly to the linked statement *in situ* at the source text.

Concept Synthesis

Figure 2 orients this explanation of how **Concept Synthesis** works. While examining a statement and its context in a scrollable window (1), the writer can pop out an automated notecard and make annotations. Each notecard records the target statement's keyword (2) and level (2). The card also records the statement itself (3). A scrollable field provides space for whatever comments the writer wants to make at this point (4). This exercise aims to consolidate ideas around central concepts introduced in the source texts. However, the exercise also serves as a *brainstorming session* in that the writer is encouraged to try out various permutations and elaborations on the core concepts. To prod thinking at this point, the writer can request a set of prompts based on known invention heuristics. Some of these are actually extended exercises that appear in an overlapping window. However, if the communication is merely a short "hint," the guidance appears in a separate dialogue box (5). While these aids foster exploration, they also focus the author's thinking.

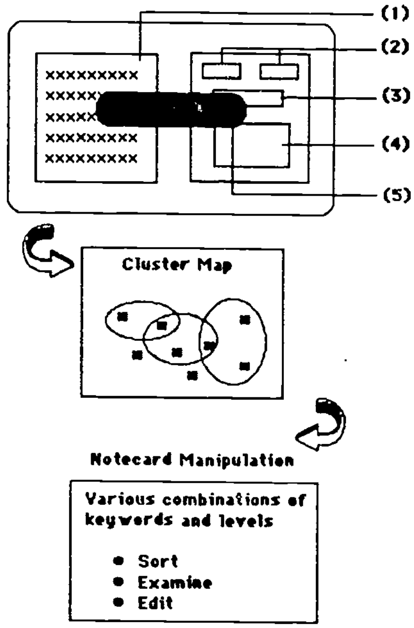


Figure 2. Components of the Concept Synthesizer

It is especially important to note that even at this relatively early stage in the knowledge design, the writer can perform two powerful operations. She can return to the cluster browser at any given time and review existing keyword maps or construct new ones -- thus viewing the writing space from a high level of abstraction. Or, she can move down to a more specific level to manipulate a set of notecards. Not only can the writer sort and filter the cards, she can try out different ordering and save each "trial run" as a separate file.

Information Threader

After a reasonable period of working with the **Cluster Browser** and its complement, the **Concept Synthesizer**, the writer may start to feel overwhelmed by the sheer amount of data and "views" generated. The **Information Threader** (see Figure 3) begins the *sculpting* process for the final text by coaching the writer to see potential structures.

The writer selects a keyword that seems to be of major importance for the source texts. Notecards from the intersection of keyword and Level 1 statements are activated, and the writer reviews the individual sentences in a scrolling window (2). Prompts aid the writer to perform mental operations such as "comparison" and "contrast" (1) or "inclusion" and "exclusion" in order to formulate an "issues" statement reflecting the concerns of these cluster-specific, top-level sentences (3). Modeling the notion of basic inquiry, this exercise leads the writer to conflate -- using the dimensions of similarity and difference (comparison and contrast) as pruning criteria. The result is a collection of possible thesis and/or topic statements.

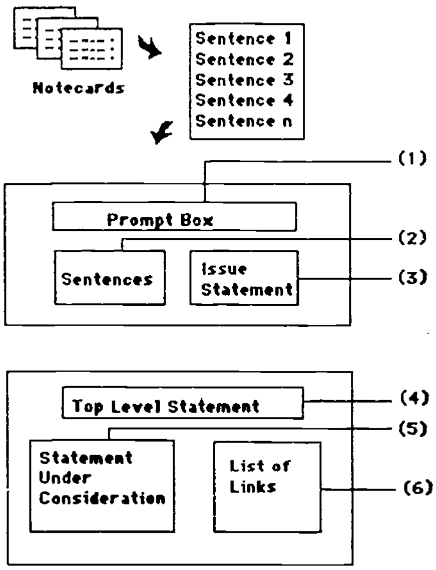


Figure 3. Components of the Information Threader

Having generated at least one "issue" statement (3), the writer moves to a second interface where the IS is displayed in a window (4). Two scrollable windows present information/data (5) and a list of possible relationships (6) to the IS, as illustrated by: *Generalizes, Specializes, Replaces, Questions, Supports, Responds to, Refers to, Objects to, Suggested by, Equal to, No Relationship* (Conklin & Begeman, 1988). The "data" window draws its input from the annotations made on the Level 2, 3, and 4 cards for the selected keyword. The writer sharpens the meaning of each annotation, prunes the excess, and assigns a relationship to the IS.

Hierarchical Planning

Writing from sources is a multi-deminsional task. Several models for composition acknowledge this complexity by positing layers of cognitive activities which the writer moves through as she hones the product in different levels of refinement (e.g. Bereiter & Scardamalia, 1987; Streitz, 1989). The **Hierarchical Planner** marks a major transition in the process modeled in the **Knowledge Board.** It is a nexus at which the information structures woven in the "thinking and threading" segment must be reconceptualized to meet the requisites of linear text. Smith, *et al.* (1987) discuss this change as a transition from a semantic net (essentially 3-D mental structure) to a hierarchical outline (essentially 2-D concrete representation). Figure 4 illustrates how this segment helps the writer to make this all-important transition from an implicit mental model to an explicit cultural artifact.

In the previous segment, the writer composed original statements that both interpreted and subsumed the intent of various clusters of Level 1 source statements. The writer also sorted through her annotations and both selected and categorized ideas relevant to the thesis/topic sentences. The result is a web of nodes (containing ideas and inferences based on sources but now in the writer's words) and links (the emergent "knowledge structure" the writer is fabricating).

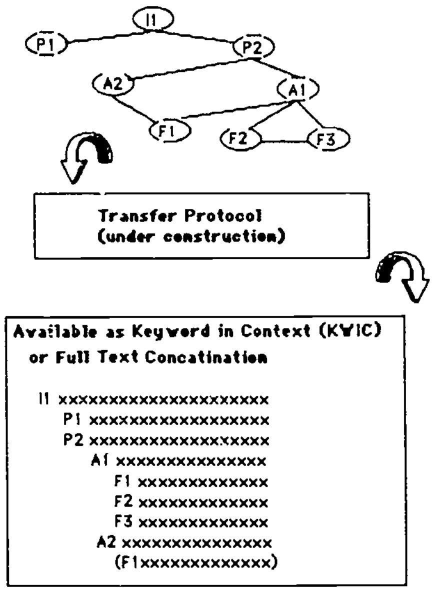


Figure 4. Components of the Hierarchical Planner

The Information Threader uses the Procedural Hierarchical IBIS paradigm (PHI) described by McCall (1990). Therefore, the web is quasi-hierarchical even at its inception. However, deriving an outline from the reticulate structure is not as simple as flattening the web. The semantic net being woven is a rich -- but tentative -- knowledge representation, mapping the "belief" structures of an author working at a given time on a given subject. Though algorithms exist for transferring semantic nets into tree structures, a formalism which simply collapses the content and divests the context from the many complex judgments and intricate decisions may be too "ham-handed" for this application. It is conceivable that a tool can be built to interpret the predicate expressions of the web. For example, if a Position has two Arguments, linked by "refutes," the entity "*_refutes* (A1, A2) might indicate a pro/con rhetoric strategy is emerging in this link.

Organization Mapping

Different theorists have suggested process models for how writing takes place. Flower and Hayes (1980) were among the first to gain national attention by using cognitive theory to account for writing. While no single theory has proven robust enough to displace all the rest, one generalization seems clear: the process is cyclical rather than linear, with certain cognitive manipulations being recursive. Such is the case with the notion of organization.

After exploring the subject domain (the source texts) and working out a richly interconnected belief structure drawn from the background texts, the writer must shift her attention to constructing a textual artifact that meets a set of external constraints and social expectations. Kooperschmidt (1985) characterizes this transition as a switch from cognitive *macro* structures to rhetoric *micro* structures.

Similar to the *drafting* stage of writing, **Organization Mapping** helps the writer to focus more intensely on the requisites of the logical form and the social conventions of text. Rhetoric and discourse studies have produced fairly detailed descriptions of the logical forms used in blocks of text (e.g., causal analysis, classification, comparison, definition, description, narration, and the like).

The interface displays two windows. One uses the concept maps and associated list of prompts to "advise" the writer. The second window displays either the KWIC version (keyword in context) of the collapsed semantic network or the full, concatenated text. The writer works not only with organizational features and with expression and stylistics but also with such situation-specific concerns as audience analysis and purpose.

Revision Heuristics

The difference between copy-editing and revision is easily characterized. Revision usually refers to more substantial changes, such as improving style, adding to or subtracting from the content, rearranging parts, or completely writing. These more global, deep-structured revision activities are associated with higher-order cognitive skills (discerning patterns in bodies of information, exercising judgment, analysis, synthesis, and other metacognitive activities). **Heuristic Revision** comprises a suite of tools for improving both coherence and expression. These tools are *strategic* (encouraging a re-thinking of high-level issues, such as purpose, point-of-view, audience analysis, voice, focus, and form) and *tactical* (including techniques of elaboration, such as level of detail, examples, support, flow, and balance).

Issue Trees: A major tool in critical-path management, this version of an issue tree visualizes the conceptual flow of the paper and the branches or decision points in the logic. This tool helps with balance by identifying choice points that are either underdeveloped or absent.

Given/New Discourse Analysis: The Prague School (among others) identifies the patterns of previously mentioned materials versus new ideas as a formalism defining inter-sentence relationships -- the essence of meaning and understanding for text. Patterns of keywords and their synonyms can be modeled using graph theory to create a tool that detects "semantic gaps" in a linear sequence of sentences. This tool also suggests alternative orderings of sentences when appropriate.

Macro-Structures: Meyer (1982) devises a taxonomy of lexical elements that signal structure in text. These "terms" (something like transitional words and phrases in traditional grammar) act as links joining chunks of content into semantically recognizable units (e.g. collection, description, causation, problem/solution, and comparison). This tool isolates cue words and then superimposes a map of links over a specified block of text displayed on the screen.

System Characteristics

Models "Effective" Behaviors: Experts are proficient at deconstruction -- that is, partitioning the task into elemental components. The expert works on the pieces for a time, steps back to compare interim results with higher-level goals, consolidates gains, jettisons unrealistic expectations or excess constraints, reorders plans (this might include satisficing), and moves back to working on the pieces again. The cycle takes place over and over during the problem-solving session. The expert excels where the novice fails because of this flexibility, this capacity to move smoothly between top-down and bottom-up strategies. In essence, the **Knowledge Board** provides an exploratory world where the writer is helped to discover (and hence to emulate) the empowering strategies of an expert writer. Like a set of "training wheels for the mind" (Carroll, 1990), the embedded tutors in this integrated environment provide balance and confidence.

Integrates Knowledge Tools: Commercial packages offering the writer a collection of tools (such as the analysis routines in the Writer's Workbench) have been around for some time now. Nevertheless, it is important to recognize that these tools are separate entities. While the writer is free to pick and choose among them, the tools are not integrated nor are they supported by AI interpreters. In other words, work done with one tool does not translate seamlessly to the "world" of another tool. At a minimum, this is inconvenient. More telling for a worker or a learner, gains in one stage of composing are not easily consolidated and carried forward to the next stage. In fact, the welter of detail generated by some tools or heuristic routines may constitute a step backwards because the writer has to deal with (1) the cognitive overload of multiple versions or even contradictory instances of the same thoughts and (2) a potentially recurrent *dis*-integration of thoughts constructed while working with different tools or heuristic devices.

Computerized writing environments are still in their infancy and can be represented by the Writing Environment (WE), developed at the University of North Carolina, Chapel Hill, (Schuler & Smith, 1990) and CSILE (Computer-Supported Intentional Learning Environments) developed at the Center for Applied Cognitive Science in Toronto, Ont., Canada (Scardamalia, 1991). Unlike the packages mentioned in the above paragraph, a writing environment's suite of tools is integrated into a rigorously structured cognitive model of writing. A well-designed environment orchestrates the writing process by emulating stages of thinking. Few existing writing environments include conventional AI applications (e.g., expert systems). Nevertheless, because the whole of the **Knowledge Board** supports and guides the activities of thinking, these knowledge-making habitats could be characterized as "intelligent."

Future Considerations

The **Knowledge Board** offers several "knowledge-weaving" paradigms, each designed to meet the requisites of a particular category of cognitive task and to exploit the talents represented by the particular user. Parts of the **Knowledge Board** "workspace" are prototyped in ToolBook™, using a set of specially created templates which "interpret" the information patterns inherent in a richly interwoven textbase. However, increased experience and development should contribute to a system which:

- Treats the "web" generated by the automatic linking feature as a semantic network.
- Encourages the users the "customize" the web though search and retrieval options.
- Incorporates an enhanced graphical interface for "browsing" the knowledge space.
- Uses AI formalisms along with the suite of templates to guide a knowledge worker in creating a synthesis from a multiplicity of sources.

A key concern is squeezing more functionality out of the basic patterns of the hyperweb environment. Adding scaffolding (more traditional forms of intelligent tutoring) at crucial transitions in the process would allow the system to monitor the individual's interaction with information, mediate among anomalies, model a domain expert's heuristics, and moderate conflict resolutions among competing strategies (thereby reducing cognitive overload by helping to manage the process).

More comprehensive theoretical research questions of interest include:

- Exploring the notion of "AI in reverse" (or knowledge tools that teach) in order to predict what new models of instruction are likely to emerge from intelligent hypertext -- e.g., the notion of a partnership rather than a master-apprentice relationship (Salomon, 1988). Unlike expert systems, hypertext is flexible, tolerant, and advisory. The system mediates between the human and the knowledgebase, while the human exercises "best judgment" in the choices. This partnership definition of "interactive" will create new decision-making and knowledge-management strategies and styles.

- Examining the relationship between spatial reasoning and literacy in virtual text environments. Because much of advanced hypertext depends heavily on the concept of information as environment, this new literacy may require the mapping of cognitive skills normally used in Euclidian space onto an information space.

References

Anderson, J.R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University.

Bereiter, C. & Scardamalia, M. (1987). *The psychology of written composition.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Carlson, P. A. & The, A. (1991). A cognitively-based neural network for determining paragraph coherence. In C. H. Dagli, S. R. T. Kumara, & Y. C. Shin (Eds.), *Intelligent engineering systems through artificial neural networks* (pp. 961-966). New York: ASME Press.

Carroll, J. B. (1990). *The Nurnberg funnel: Designing minimalist instruction for practical computer skill.* Cambridge, MA: MIT Press.

Conklin, J. & Begeman, M.L. (1988). gIBIS: A hypertext tool for exploratory policy discussion. *ACM Transactions on Office Information Systems*, 6(4), 303-331.

Flower, L. & Hayes, R. (1980). The dynamics of composing: Making plans and juggling constraints. In L. W. Gregg and E. R. Steinberg (Eds.), *Cognitive processes in writing* (pp. 31-50). Hillsdale, NJ: Lawrence Erlbaum Associates.

Kopperschmidt, J. (1985). An analysis of argumentation. In T. A. van Dijk (Ed.), *Handbook of discourse analysis*, Vol 2: *Dimensions of discourse* (pp. 159-168). London, UK: Academic Press.

McCall. R., Bennett, P., d'Oronzio, P., Ostwald, J., Shipman, F. & Wallace, N. (1990). PHIDIAS: A PHI-based design environment integrating CAD graphics into dynamic hypertext. In N. Streitz, A. Rizk, & J. Andre (Eds.), *Hypertext: Concepts, systems, and applications* (pp. 152 - 165). Cambridge, UK: Cambridge University Press.

Meyer, B.J. F. (1982). Prose analysis: Procedures, purposes, and problems. In B.K. Britton & J. Black (Eds.), *Analyzing and understanding expository text.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Salomon, G. (1988). "AI in reverse: Computer tools that turn cognitive. *Journal of Educational Computing Research* 4(2), 123-139.

Scardamalia, M. (1991). An architecture for social construction of knowledge. *Proceedings of the International Conference on the Learning Sciences* (pp. 397). Evanston, IL: Association for the Advancement of Computing in Education.

Schuler, W. & Smith, J. (1990). Author's argumentation assistant (AAA): A hypertext-based authoring tool for argumentative texts. In N. Streitz, A. Rizk, & J. Andre (Eds.), *Hypertext: Concepts, systems, and applications* (pp. 137-151). Cambridge, UK: Cambridge University Press.

Smith J. B., Weiss, S. F., & Ferguson, G. J. (1987). A hypertext writing environment and its cognitive basis. *Proceedings, Hypertext '87 Conference* (pp. 195-214). Chapel Hill, NC: Association for Computing Machinery.

Streitz, N. A., Hannemann, J. & Thuring, M. (1989). From ideas and arguments to hyperdocuments: Traveling through activity spaces," *Proceedings, Hypertext '89 Conference* (pp. 343-364). Pittsburgh, PA: Association for Computing Machinery.

TEAMSS: Enabling Middle School Teachers to Use and Repurpose Interactive Videodisks in Science Classrooms

RACHELLE HELLER
C. DIANNE MARTIN
Department of Electrical Engineering and Computer Science
The George Washington University
Washington DC, 20052, USA

New interactive multimedia materials are presented in conferences and professional publications as having the potential to revolutionize if not just revitalize, k-12 education (Beeman et al, 1987). The opportunity to provide dramatic, clear images or videos as part of a classroom experience and the ability to order them according to users preferences and to intertwine them with other computer augmented instructions seems to offer a amazingly rich educational environment.

Unfortunately much of this material is offered to teachers who were educated in the BC (Before Computers) generation (Heller & Martin, 1984). Even though there have been many inservice and preservice programs to train teachers how to use the new technology, (Bitter & Yohe, 1989), research has shown that these programs have not been totally successful in bringing about real change in the classroom (OTA, 1988; Criswell, 1989; Martin, 1988) Studies looking at change related to computer use in the classroom by teachers indicate that teachers need to be presented material appropriate to their level of concern if they are to institutionalize it into their teaching practice in any substantive way (Heller & Martin, 1987)

Teachers must move at their own pace through stages of concern that are first *self-directed,* then *task-directed* and finally, *impact-directed* (Heller & Martin, 1987). Self directed concerns deal with the effect of the innovation upon self. In the case of computer technology in the classroom, teachers have expressed concerns that they would be unable to learn the technology, they would be replaced by the technology, or their students would know more about computers than they they would. Task-directed concerns deal with innovation as it related to a particular environment. At this level, teachers would start to focus on how they might actually use and manage the technology in their particular classrooms. It is at the level of impact-directed concerns, however, that real change can take place. Those concerns deal with the impact of the innovation across a set of environments, such as a school or district. Teachers at that level of concern are interested in collaborating with colleagues to investigate how the innovation can be refocused and refined to meet diverse needs across the curriculum.

Teachers involved with any educational innovation will go through these progressive stages of concern, and such concerns must be taken into account during teacher training is it is to be successful. Clearly, not all individuals are at the same stage of concern at the same time. This lack of uniformity of concern among a set of participants at a workshop will cause the same workshop to be a highly successful experience for some, while others may view the same workshop as frustrating or inappropriate for their needs,. This can be true even if the participants come from the same teaching population and teach the same kinds of students. Thus, when looking at a new technology, such as the use of interactive videodisk, it is important to be sensitive to the varying needs and concerns of the participants when designing training for them.

In addition to the fact that individuals have different needs concerning the use of interactive videodisk in their classrooms, they also differ in their ability to effect change in their school or school system. For example, a classroom teacher who returns from a workshop full of enthusiasm and ready to incorporate the new material into the classroom may not have the knowledge necessary to work through the administration system and red-tape to obtain the necessary hardware and materials in order to utilize the workshop information. As a novice teacher, they may not have the teaching experience to see how to best use the information in their school setting. Often only one person from a specific school or school setting attends an workshop and once back in the home school that person has to deal with all the aspects of incorporating an innovation into the system. This lonesome situation can deter many enthusiastic teachers from following through with the material. Even if there is a supportive administrator, the administrator rarely has deep knowledge about the innovation and their commitment to the project is frequently not at the same level as the workshop participant.

Based on these aspects of individual differences - stages of concern and familiarity with the system - and the need for support once a workshop is over, we developed an in-service, year long program to work with middle school educators as they learned to use and repurpose interactive videodisks in their science classrooms. The title of the program, The TEAMSS Project: Teacher Enhanced Applications for Middle School Science, underscores our belief that it is the teacher/educator and not the technology that enhances classroom learning opportunities and that workshops that empower teachers are the most effective.

Workshop Description

Audience

This program was designed to appeal to teams of teachers/educators from the same school or school system. Each team consisted of three members: a novice teacher, a master teacher and an administrator. Each team member came to the program with specific strengths designed to predict success for the team. The novice teacher is someone who has less than 5 years teaching experience in his or her current discipline. This category includes someone who has just entered the teaching profession as well as someone who is now in a new teaching setting, not strictly identifiable with their training. Very often in middle school, to meet the demand for science teachers and demands of the unions, science teachers are assigned from amongst teachers who do not have science education backgrounds or even backgrounds in the liberal arts. The novice teacher brings recent academic training to the team at best and enthusiasm for teaching and an appreciation of the challenge of new materials at least. The master teacher is someone with many years of experience in the present science classroom. Rarely is the master teacher rewarded for excellent teaching by being given the opportunity to teach other teachers. In fact, when master teachers are identified they are often rewarded by being moved out of the classroom into administration positions, thereby curtailing their contact with other classroom teachers. The master teacher, therefore, brings classroom experience and insight to the team. The final member of the team is an administrator. This person has a broader view of the educational process in this school or school system. Administrators rarely get opportunities to work closely with those teachers who report to them. Further, typical administrator workshops, especially about technology, usually only address the scope of the technology and rarely go into depth on its use or application. The administrator brings to the team the ability to work through the system and to "make things happen."

Each of the three types of participant also had something unique to gain from this experience. The novice teacher can gain both specific content information about using computers and interactive videodisk for science education as well as establish new connections in the professional networks of teacher/educators that can be called upon for guidance. The master teacher can gain recognition for good work and a 'shot in the arm' from exposure to new classroom materials. The administrator can gain reinforcements about the personnel in their charge as well as deep information about the use of videodisk in the classroom.

In order to maximize the networking aspect of this workshop, groups of teams into two larger groups and travelled through the workshop together. The two cohorts met during certain all group activities such as weekly bull sessions and guest lectures. Figure 1 shows the demographics for the participants in the first two years of the project. All participants receive three graduate credits, a summer stipend to defray lost wages from workshop participation and a mini-grant to enable participants to attend a conference of their choice. In addition, team members who live too far from the summer program are offered room and board.

Position	1991 Participants	1992 Participants
Novice Teacher	19	10
Master Teacher	17	18
Administrator	7	8
Peer Mentors	—	3
Number of multilevel teams	12	15
Number of individual projects	1	2
Average years teaching	12.0	14.5
Average years computing	2.8	4.0
Computer usage per week		
school	4.4 hours	6.5 hours
out of school	6.6	4.3
student use	2.7	3.8

Figure 1: Participant demographics

Program

The year-long workshop consists of three phases: a four week summer program, a fall follow-up program and a spring follow up program. The summer schedule (Figure 2) consisted of three strands: a computer skills strand, a videodisk skills strand and a cooperative education/content strand. The timing of these strands was rotated for each group of teams each day to insure that all participants got to work on specific strands during their most productive time of day at least some times during the week.

In the video laboratory participants had access to Pioneer 2200 videodisk players connected to Macintosh computers. Each station had a 13 inch video monitor for the player and one station had a 27 inch video monitor to be used as a demonstration station. While the intent of the workshop was to enable teachers to use and re-use interactive videodisk (level 3 videodisks) in their classroom the first task in the video lab was to introduce the teachers to videodisks (level 1 videodisks) and how to use them. Teachers had to be shown how to operate the player, how to open, load and close the player drawer and how to play a video sequence.

One of the most frustrating aspects of using an new material in the classroom is the lack of time teachers usually have to really study the material and become familiar with the contents. To address this issue while increasing the teachers skills using the videodisk player, we began by using the remote control device and created a series of scavenger hunts in which the teachers were to find a variety of facts or features on one of the disks from the Smithsonian Institute. Teachers were asked to report the frame number in which they found the answer to a specific question and they were also asked to identify sequences of frames which they might be able to use in their classroom and why. This activity took more than two hours but it allowed each team to become completely familiar with the contents of one of the disks. We specifically chose the disks from the Smithsonian because they were inexpensive and we felt that the teams could purchase them if they found one they liked. In addition, our proximity to the Smithsonian and to the collections made it possible for teachers to plan on using one of these disks before a class field trip.

Beyond having the time to become familiar with the specific content of these disks, we wanted to encourage teachers to find ways of incorporating the material into their existing way of presenting materials in class. In addition, we wanted to emphasize how easy it might be to extend their current lessons with videodisk material. To accomplish this, the second major video lab activity centered around using the bar code reader. At first we only used the barcoder with those disks that came with a barcode catalog (eg BioScience). We then gave each team a barcode creation program (compliments of Pioneer). This allowed teachers to make barcodes for sections of materials they found interesting. We encouraged teams to take the textbooks that were currently in use in their home classrooms and create a series of barcodes to underscore a section in those texts.

In preparation for writing their own computer interface to videodisk material, we wanted teachers to become familiar with many of the existing presentations and metaphors for presentation. There are a great many level 3 videodisk packages already developed and available to schools. These usually include a videodisk, a text of materials and some type of computer interface to enable students and teachers to use the video material in a hypertext type fashion. In the next activity, teachers reviewed the computer based videodisk material prepared by many publishers. We identified one level 3 videodisk (Visual Almanac by Apple) because of its sampler nature and urged all teachers to review the material on this disk even though some of the material dealt with curricula outside the sciences. We wanted teachers to see the breadth of possibilities of topics to cover and presentation techniques and begin to consider how they might structure their own interactive videodisk presentation.

By this time, the teachers were feeling rather confident in their skills related to using the videodisk player and the computer. While the video lab materials were being presented, the teams were also working on their Hypercard skills. By this time in the project, they were ready to combine their computer skills and their video skills to create their own projects which re-used the videodisks in ways they felt suited their particular classroom. Choosing a topic turned out to be the most difficult aspect of the 4 week workshop. Teams met with the principal investigators to define and refine their topics and to discuss the development and presentation. Once the projects were defined, there was a team presentation for peer review. All of this was intended to short circuit the development time and enable teams to finish a project in the remaining two weeks. Once the projects were completed, the teams presented the working version to the other participants. Some of the projects were also shown at the workshop banquet to which their supervisors were invited.

Once the workshop was over, each team was expected to bring the workshop information back to their particular school or school setting. The fall follow up program was designed to help teams plan an in service day for their own school and to review the successes and failures that each team had experienced in trying to establish interactive videodisk in their science classrooms. The spring follow up program, which was created in response to the participant's request, focused on skills needed for computer based image capture.

Examples of Materials Developed

As noted in Figure 1, there were a total of 27 group projects and 3 indep dent projects developed by the participants over the two years. During the 1991 session the focus of the ing was intended to be on life sciences and in 1992 the focus was on physical sciences. The majority of the projects reflect this dichotomy. Typical topics of the interactive programs developed include light, simple machines, sound and weather in the physical sciences and insects, the food chain, plants, seeds structure, and dinosaurs in the life sciences. There were a few topics which could not be categorized into either physical or life science: model or theory development, science careers and the role of black scientists in the history of science.

The design of almost all of the projects was similar. They began with some sort of title or attract-screen (see Figure 3) and followed this with a teaching sequence (figure 4) and finished with a test sequence (figure 5). During the teaching, they often showed a short section of videodisk material as expletive of the topic described on the computer screen. For example in the project on the food chain (figure 4) once textual material was presented concerning producers and consumers, students could click on an on-screen button and the video monitor would show a clip of typical producers in the food chain.

ERIC

Week 1:	9:00 to 10:30;	10:30 to noon;	noon to 1:00;	1:00 to 2:30;	2:30 to 3:45;	3:45 -
Day 1	Introduction to project		lunch	computer skills check		daily
Day 2	ML	VL	lunch	CL	SC	
	macpaint macwrite	introduction to videodisk				eval.
Day 3	ML	VL	lunch	CL	SC	
	Introduction to Hypercard	Using barcodes				
Day 4	ML	VL	lunch	CL	SC	
	Hypercard	Barcoders			Bull Session	
Day 5	ML	VL	lunch	CL	SC	
	Open Lab	Open Lab				

Week 2:	9:00 to 10:30;	10:30 to noon;	noon to 1:00;	1:00 to 2:30;	2:30 to 3:45;	3:45 -
Day 2	ML	VL	lunch	CL	SC	daily
	Hypercard buttons	Computer Control				
Day 2	ML	VL	lunch	CL	SC	eval.
	Hypercard backgr., fields	Computer Control				
Day 3	ML	VL	lunch	CL	SC	
	Hypercard backgr., fields	Computer Control				
Day 4	ML	VL	lunch	CL	SC	
	Comptrs in Ed.	Open Lab			Bull Session	
Day 5	ML	VL	lunch	CL	SC	
	Hypercard scripting	Hypercard scripting				

Week 3:	9:00 to 10:30;	10:30 to noon;	noon to 1:00;	1:00 to 2:30;	2:30 to 3:45;	3:45 -
Day 1	ML	VL	lunch	CL	SC	
	All day guest demonstration and lecture by Tom Boudrot, Videodisc producer					
Day 2	ML	VL	lunch	CL	SC	
	Laptop demo	Hypercard control				
Day 3	ML	VL	lunch	CL	SC	
	HyperCard variables	Hypercard linking stacks				
Day 4	ML	VL	lunch	CL	SC	
	Open Lab	Lego/Logo			Bull Session	
Day 5	ML	VL	lunch	CL	SC	
	Open Labs	Peer review of proposed project				

Week 4:	9:00 to 10:30;	10:30 to noon;	noon to 1:00;	1:00 to 2:30;	2:30 to 3:45;	3:45 -
Day 1	ML	VL	lunch	CL	SC	
	Open Labs	Evaluation of Software lecture				
Day 2	ML	VL	lunch	CL	SC	
	Open Labs	Open Labs				
Day 3	ML	VL	lunch	CL	SC	
	Open Labs	Open Labs				
Day 4	ML	VL	lunch	CL	SC	
	Open Labs	Project Presentations			Bull Session	
Day 5	ML	VL	lunch	CL	SC	
	Project Presentations			Final Evaluation		

Figure 2. Typical summer workshop group schedule

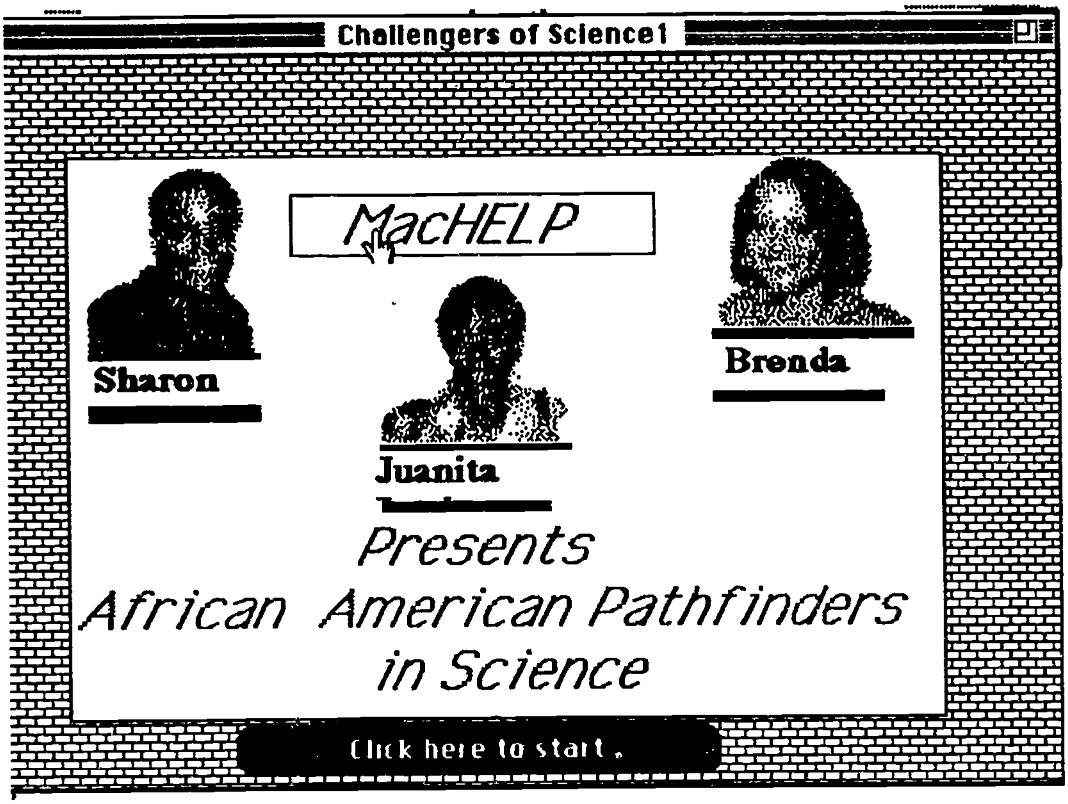


Figure 3: Opening Screen

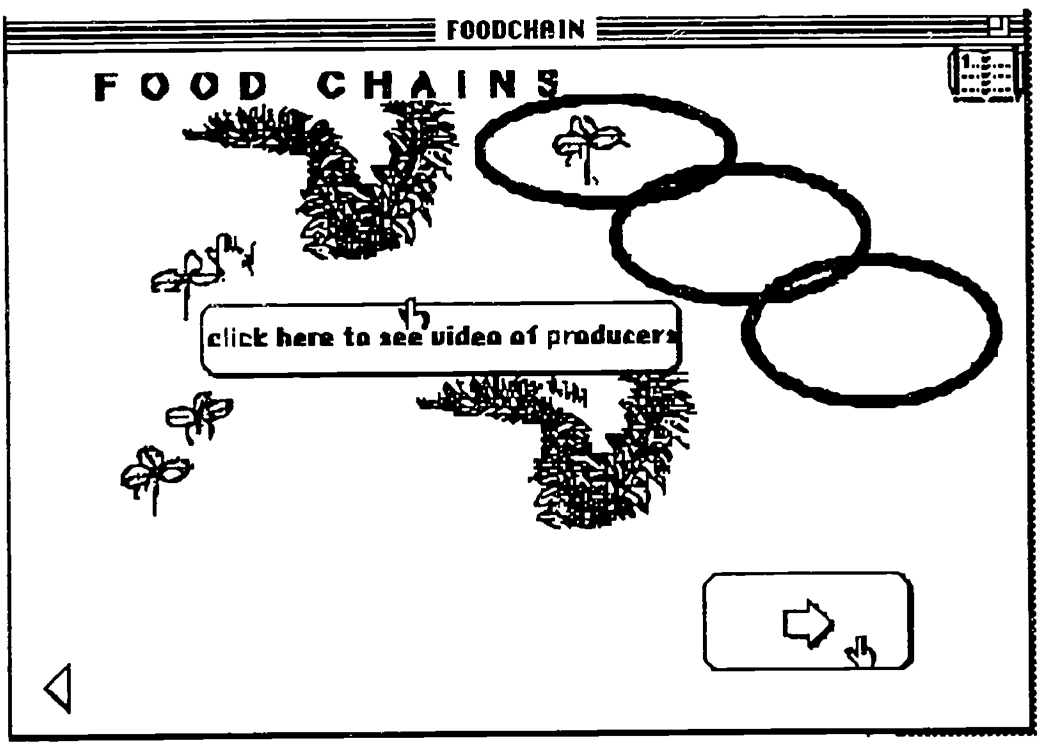


Figure 4: Presentation

BEST COPY AVAILABLE

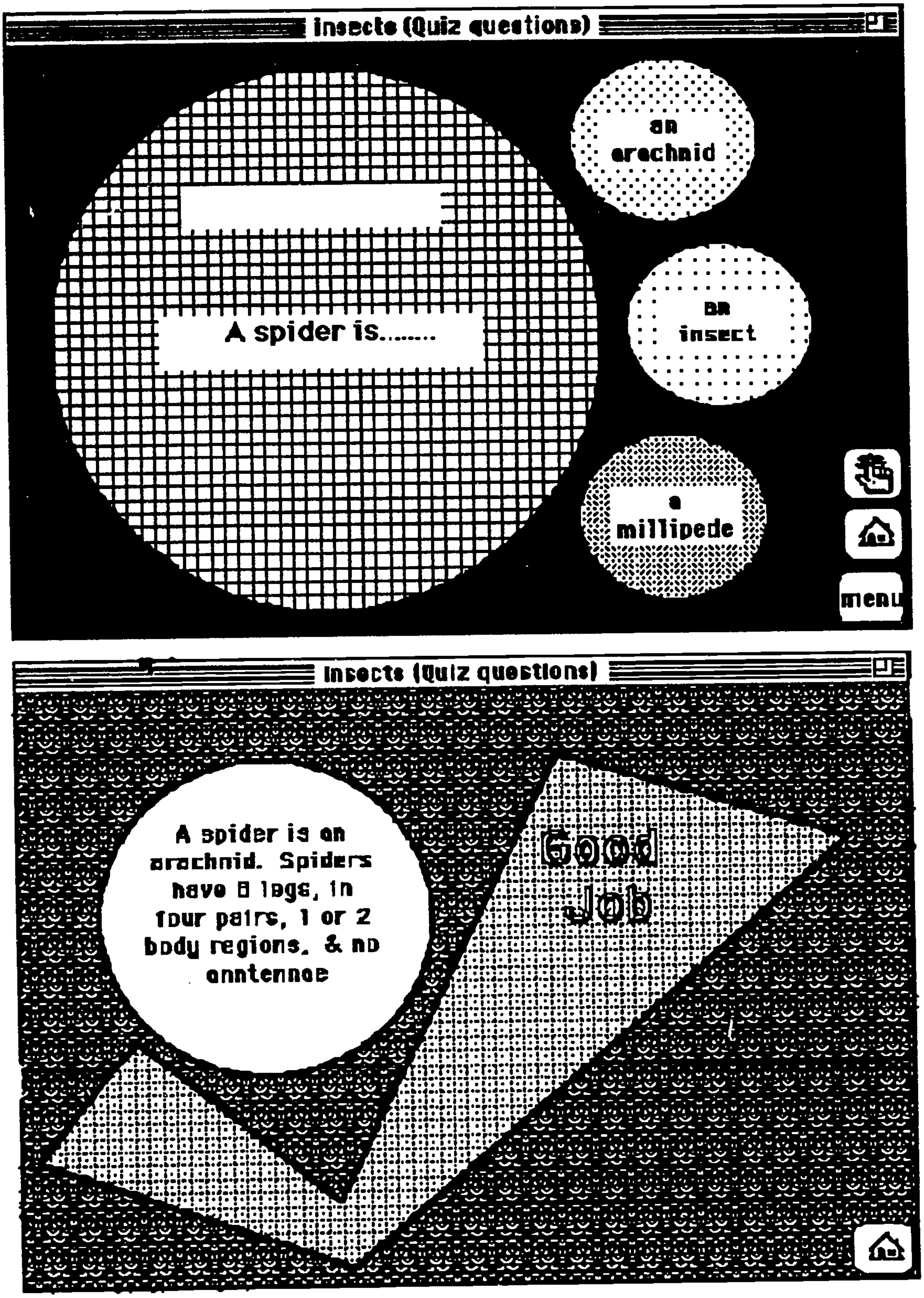


Figure 5: Questions

BEST COPY AVAILABLE

Evaluation of the Workshop

Participants responded to daily and weekly evaluations as well as evaluations at the end of the program. The anonymous reports indicated how well a respondee felt he or she was doing in relation to the content being delivered. It also asked for a report on how well the team was functioning as a team. These reports helped the project leaders to make microscopic changes to the workshop on a daily basis to meet the individual needs. The weekly 'bull session' was held every Thursday. The concerns reported at these meetings were often able to be addressed before the weekend insuring that the participants felt the workshop was very responsive to their needs.

Participants also recorded in a learning log in which they were encouraged to reflect on their own growth. It was a place for the teachers to reflect on how they were being taught, how this new information fit into their current scheme of things and how they might use this new information. At times, specific questions were posed to the participants for journal reflection and at other times they were encouraged to write freely. Any pages that they did not wish read were to be stapled together.

The evaluations held after the summer workshop were used to help plan the content for the fall follow-up. In fact the spring follow-up event was created in response to team members' requests for additional material and contact time.

Findings

All teachers reported confidence in using the videodisk player (Heller & Martin, 1992) and felt that they could present this material in their classrooms and to other teachers in their schools without the help of the workshop principal investigators. For some this really meant only being able to use the videodisk player with the remote control. For the majority of the participants this meant being able to use already produced materials. For a small group, this meant being confident that they could create, on their own and outside of the workshop environment, materials using Hypercard to control and present videodisk content.

The informal debriefing at the follow up sessions identified three specific and common problems faced by teacher/educators as they returned to the classroom. The first was the lack of equipment. Even though each TEAMSS participant had the commitment of an administrator to insure that the participant would have access to the equipment before the participant was accepted to the program, finding hardware, making it work and locating a specific videodisk was an almost universal problem. Typically, the teams from larger school systems faired worse in establishing access to equipment. Very often the administrator who promised the material was no longer the administrator to whom the teacher/educator reported. The most successful teams were those from small (or private) school systems or those whose senior administrators had visited the program during the summer's final banquet at which participants demonstrated their projects.

The second most common problem that the participants faced centered on finding the time to share their knowledge with others in their school setting. Most team members were immediately identified as a local school resource and many of their peers wanted to have them demonstrate interactive videodisk materials in their classroom. Team members reported that they used time they had set aside to further their own videodisk use in order to help colleagues.

The final problem faced by teams was the lack of other programs to help them advance their own education in the new media. By the end of the second year after the completion of the program, many team members were asking about advanced workshops in such areas as videodisk mastering and the use of CD ROM materials.

The most curious aspect of this project is the type of materials that the teams chose to develop. Even though the project offered many models of interactive videodisk, including database surveys, presentations, games and tutorials, almost every project had a test phase as part of the material. Does this mean that when pressed, teachers see the simplest model of education as one which presents materials and then tests the student's comprehension? Are teachers so acculturated that they are constantly 'teaching to the test"? When asked, these teachers indicated that they never give a test immediately after presenting a topic and certainly not in the same lesson. They present a topic, review it and then, perhaps, give a quiz or test. On the other hand, they do use questions and other clues to decide how much learning is going on on the part of the students. It may be the lack of external clues using the computer based materials that prompted teams to include the test section. However, even if the tests were an attempt to simulate this external clue teachers use to to determine whether learning is taking place, it is still curious that teams chose the anxiety-producing term "test" to identify this section in their program.

This rather traditional use of new technology suggests that most of the teachers/educators and administrators in the workshop are still at the task-directed stage of concern in relation to videodisk technology. They were still attempting to make it fit into their regular teaching paradigm rather than re-examining and redefining the paradigm to realize the full potential of the new technology. In addition, this emphasis upon the traditional tutorial followed by testing raises concerns in light of the research that suggests that the kind of computer programs children are exposed to send subtle messages to them about their self worth, sense of control and computer capabilities (Becker, 1990, Martin, 1991)

Suggestions for Future Programs

The basic program design utilizing team participation and cooperative learning seems to have been successful and should form the basis of future programs.The three most common problems identified by the teams participants: lack of equipment, high demand on the 'facilitators time, and lack of programs building on the workshop skills, are a good starting point for developing new content areas. Based on interest expressed by the teachers, the first content change should focus on including videodisk mastering or other methods of collecting and presenting primary images.

Future programs should include reflection on how teachers perceive the role of computers in their classroom and how they might harness computer power to their own ends. The fact that so many projects included a testing phase leads us to suggest that teachers need time to look critically at the pedagogic message of computer materials as well as the information content. Only when these two spheres are addressed will computer material become integral to the teaching that takes place in the classroom.

References

Becker, H. (1990) "*When powerful tools meet conventional beliefs and institutional constraints: National survey findings on computer use by american teachers.*" Report No. 49, Center for Research on Elementary and Middle Schools, Baltimore, MD.

Beeman, W.O., Anderson, K. T., Bader, G., Larkin, J., McClard, A.P., McQuillan, P., & Shield, M (1987) Hypertext and pluralism: From linear to non-linear thinking. In *Proceedings of the ACM Hypertext '87 Conference*, Chapel Hill, NC. November 13-15, p67-68

Bitter, G & Yohe, R. (1989). Preparing teachers for the information age. *Educational Technology, 29, 22-25.*

Criswell, J.R. (1989) Rethinking microcomputer instruction as part of teacher education reform. *Educational Technology*, 29, 40-43.

Heller, R.S., & Martin, C. D. (1984). Computer Literacy for the BC Generation. *T.H.E. Journal*, January.

Heller, R.S. & Martin, C.D. (1987) Measuring the Level of Teacher Concern over Microcomputers in Instruction. *Education & Computing: The International Journal*, #(3,4).

Heller, R.S. & Martin, C.D. (1992) TEAMSS: Staff development for middle school science using hypermedia. *Technology and Teacher Education Annual*. Association for the Advancement of Computing in Education, Charlottesville, VA., March

Martin, C.D. (1988) School District Implementation of Microcomputers for Instruction, *Journal of Research on Computing in Education*, Winter

Martin, C.D. (1991) New Findings from Qualitative Data Using Hypercard: Microcomputers, Control and Equity. *Computers and Education Journal* 16(3), Pergamon Press

U.S. Congress, Office of Technology Assessment (1988). *Power On!: New Tools for Teaching and Learning*. OTA-SET-379. Government Printing Office, Washington D.C.

Credits

The work reported in this paper has been supported by a grant from the National Science Foundation: TPE 9055370

"La Plaza" A Software Design for an Educational Network

PEDRO E. HEPP, MARIA I. ALVAREZ, ENRIQUE HINOSTROZA, ERNESTO LAVAL
Departamento de Ciencia de la Computación
Pontificia Universidad Católica de Chile, Casilla 306 (143) Correo 22, Santiago, CHILE
y
Programa MECE, Ministerio de Educación Chile

A major effort in education is under way in Chile. The World Bank and the Chilean Government have started a 5 year Program in 1992 whose key objective is to enhance the efficiency, quality and equity of primary education in selected schools in targeted urban and rural low-income areas. Part of this initiative is a project that will explore the role of computers and communications in a sample of about 100 primary schools in a few Chilean regions. The project will be evaluated every year in order to eventually expand it to the whole Chilean educational system. Three major aspects were considered during the project's design:

1. The Chilean educational reality: its high degree of administrative centralization, a large number of rural and isolated schools, poorly motivated teachers, school abandonment, low educational performance indexes, few opportunities for local innovation, etc. The cultural, geographical, social and ethnical diversity was also considered (Muñoz, 1988).

This reality offers new opportunities for computer uses. Basically, computer networks may reduce social and professional isolation, they can be used for collaborative projects among teachers, students and other community members, they can enhance people's world vision and they can also be used as a tool in educational projects (Fisher & Lipson, 1984).

On the other hand, this heterogeneous reality imposed a careful design of the way the computers would be introduced at each school. At each site, the role, the value and the uses of the computers may vary. The size of the project, about 100 schools, was determined in order to cover a comprehensive part of this reality. Thus, if the project succeeds, it should be possible to expand it to cover more of the same.

2. The world wide experience in computers in education (O.T.A., 1989) and its status in Chile was revised in order to assess other large scale efforts and to look for trends in usage, training, technology and evaluation mechanisms. The last decade was regarded as one with high expectations and few significant results in terms of the role, value or effectiveness of computers in schools (Hirschbuhl & Konet, 1990). In Chile almost 30% of schools in Santiago (which has 40% of the population) have old 8 bit computers and very few software. Teachers are mainly self-trained and the computers are basically used in extracurricular activities for teaching Basic and Logo (Sanchez, 1991; de la Puente & Araya, 1991).

A redefinition of the role of the computer in education as a means and not as an end for the learning process, was the result of this analysis. This redefinition seems to be a strong trend in new projects of computers and education (McAnge, Harrington & Pierson, 1990; Ruth & Ronkin, 1992).

3. The new opportunities of modern technology. Computers of the nineties include enhanced capabilities allowing new ways for using them in schools. In particular multimedia -integrating text, sound, animation and live video in a multidimensional associative space- and communications (Park, 1991; Ambron & Hooper, 1990; Nelson & Palumbo, 1992; Gillete, 1992), for

ERIC

collaborative projects among geographically dispersed schools (O.T.A., 1989). In this respect, the local communication companies are rapidly improving and expanding their telephone systems, and some are installing large fiber optics networks. It is also important to note that the new available computing capabilities, such as the ability to run short video sequences with no special hardware on small machines, were found a few years ago on only high-end workstations. This fast evolution led us to design for the future, and to use today's best technology.

These three considerations led us to propose two main goals for the project:
1. Starting in 1992, gradually develop a computer network comprising primary schools and research centers in four Chilean regions, using modern computers and the public telephone system. Foreign schools should be linked using the Internet facilities as a gateway. The network should gradually expand until it reaches about 100 schools covering a variety of cultural, geographical, social and ethnical communities. It is also expected that at least 10 research centers (universities and institutions) related to education and community development initiatives will join the network to participate in collaborative projects. The network has a hierarchical architecture, with national, regional, community and school nodes, and it operates asinchronously, once or twice a day and mainly during the night. This mode of operation, having most of the traffic over night, is due to cheaper traffic rates and because of the availability of the telephones in the schools.

It is important to emphasize the effort on providing very simple to use human-interfaces, building an abstraction over the operating system and the communication platform. Training on how to become a computer user and how to take advantage of the communication system and the educational software has a low profile. The experience conducted so far demonstrates that carefully chosen graphical metaphors can dramatically reduce computer training time. Training is being devoted to methodological rather than to technical issues, i.e. we will not concentrate on teaching about computers in this project. The aim is to provide tools for collaborative educational initiatives among teachers and students.
2. Install two "Educational Technology" Laboratories, one in Santiago and one in a regional university to support, design and stimulate the activities that will be carried on through the network. These laboratories will have the resources, the people and the equipment to build and test multimedia educational software. They are being staffed by an interdisciplinary team, comprising software and communication engineers, teachers, psychologists, graphics designers and social workers.

The Project's status

The Project started in early 1992, but in 1991 a prototype network was implemented in Santiago comprising 6 nodes: 3 local primary schools, the Catholic University of Chile, Apple Computer headquarters in Santiago and a school in Albuquerque, New Mexico. The schools in Santiago were 2 public schools in low-income areas and one private high-income school.

The lessons learned during the experience of 1991 led to a redesign of the user interface of the software. The concepts that support it and the communication system were validated with this first experience. In 1992 the network has expanded to include 8 nodes in Chile and 3 foreign schools.

The hardware platform used in Santiago is presently Apple Macintosh LC II. Due to its gradual approach, the project is initially installing only one computer-modem-printer kit plus software at each school. In the long term, each school will receive at least two computers, according to demand. Although these numbers are small, it is important to consider the following:

- only primary, mainly small, one course pre level schools are presently included in the project.

- one-computer-one-student is not a long term goal, not only because of costs but also because of some facts, like the effectiveness of team-work, the importance of having most of the learning experiences outside a computer screen and the use of the computers mainly as tools needed at particular times or as part of educational projects that occur mainly without computers.

- the project aims to prove its effectiveness on a small scale before going national. More experience is needed to properly assess the cost/benefits of using computers in Chilean low-income schools.

Software Design

The first issue addressed during the software design was the end user: typically a 8-12 year old child or a teacher in a low-income area school. Both kind of users are well defined in psychological studies that describe the characteristics of the "The Culture of the Poverty" (Moffat, 1972; Bravo & Montenegro, 1977).

This reality imposed some special requirements on the software. Some of these are:

- Familiarity. Considering that children come from an environment where learning follows a fundamentally oral and visual tradition, the graphical notations, icons, metaphors and sounds used in the software were of utmost importance to provide familiar, non intimidating user interfaces. In this respect, hypermedia technology offers ample room for innovation.
- Relevancy. The software content should be designed according to what is important to the users. i.e. the particular characteristics of their living environments, their hopes, immediate needs and expectations. For example, a software to be used to learn or exercise with arithmetic, should consider that many low-income urban children work by selling goods at the streets or at their parent's shops. However, children living in rural setting use arithmetic mainly for counting sheeps, horses and cattle (Arancibia & Roa, 1986).
- Inspiring. The software should not intend to become the protagonist of the child's education. Rather, it should relate to the student's needs and knowledge and suggest observations, learning activities and projects to be performed outside the computer screen.

Another issue in the design was our lack of knowledge about the particular characteristics of each participant of the educational system and the special needs that we will certainly find during the implementation and execution of the project. So we decided to provide a very flexible software platform that could be easily modified.

The result of this design was an environment that represent and simulates different elements of a town. At this stage of the project we provide a Post-Office, to interchange e-mail, a Museum where different educational software can be found, a Kiosk, with pertinent news and short stories and a Cultural Center where collaborative projects are initiated and coordinated.

Software Implementation and Experience: The Plaza

The first metaphor the user finds on the computer screen is a Plaza. Plazas are well known places in Spain and Latin-American towns and cities. A Plaza is an attractive place for children, used for getting together and to start interesting adventures. Figure 1 is a black and white version of the full-colored original interface. The software is presently a prototype built with hypercard, Micromind Director and other software tools.

There are no command languages involved in using the Plaza, nor is it necessary any particular training, only pointing and clicking with the mouse. Its main purposes are to be inviting and to offer a simple window to the available functionality. Teachers and students were exposed to a short demonstration and were then left on their own. In most schools, the computers were first left to the teachers for a few days in order to allow them to get confident with the system and prepare them for trouble shooting. Most problems and questions were solved using the e-mail system (the Post Office) itself without having to go to the schools.

The Plaza has several components, each one being a well known metaphor to the user, with a particular meaning and purpose: a Kiosk, a Post office (Correo), a Museum and a Cultural Center (Centro Cultural).

The Kiosk

The Kiosk offers a window to an information space comprising newsletters, stories and educational comics. Student's newsletters have a lively format and include news provided not only by a central editor but also by the students themselves who send their contributions vie e-mail. It is expected that the student's newsletters will gradually evolve into several local and specialized newsletters in the network, most of them built by the children themselves. Similarly, the teacher's newsletter has for now a central editor and contains educational news and also general purpose articles, such as the project's advancements, global educational news and trends. The newsletters include presently text and graphics, animation's and short video sequences.

The Kiosk includes also short stories and comics as a stimulus for reading. Every time while reading the newsletters, comics or stories, students are encouraged to send their own illustrated comments or stories which, after some central editing, can be found in the Kiosk.

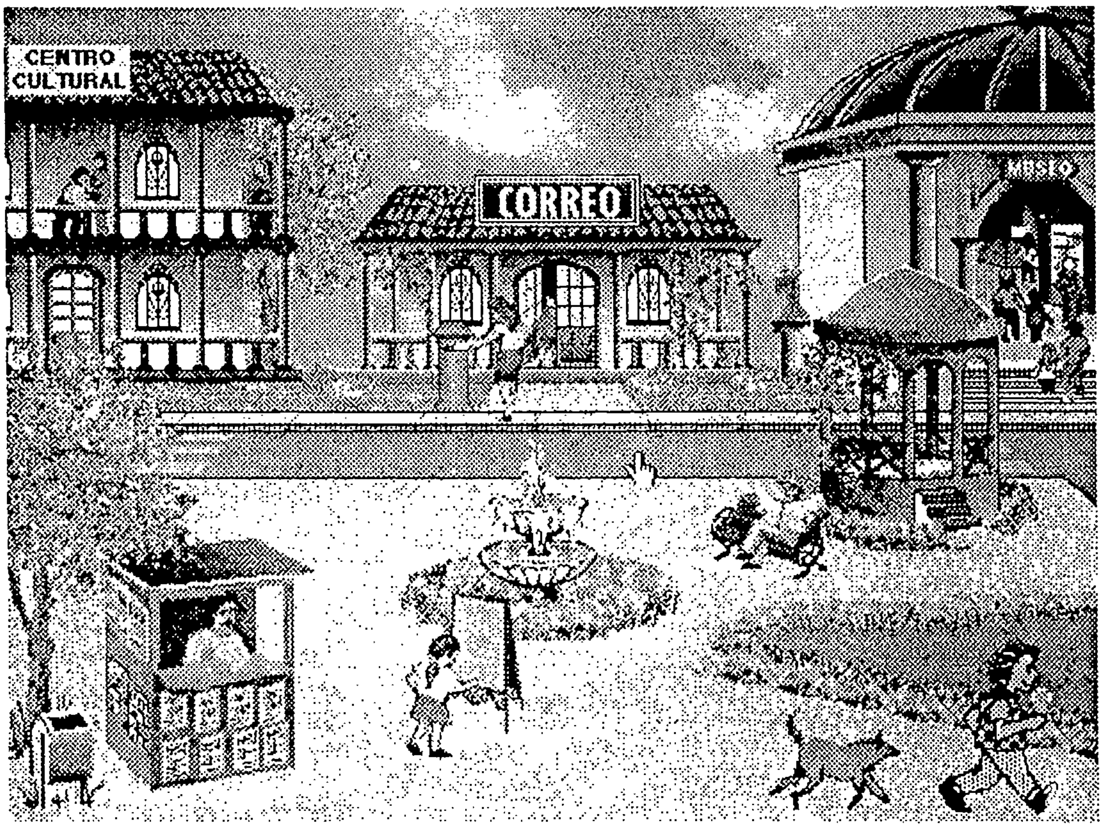


Figure 1. A b/w version of The Plaza

The Museum

The museum is a friendly interface for a simple educational software database. Users can browse through the museum and select a piece of software. If the selected material is on place, it is run for the user -generally a multimedia educational application-, if it is not, a message is sent to the central server in order to deliver the material the next day.

Presently, only four applications are available at the museum but one of the project's goal is to gradually build a comprehensive library of tested software. Most of the software is being built in collaboration with teachers. Only proven, robust and well designed applications are

BEST COPY AVAILABLE

allowed into the Museum's collections. Each application has a "curriculum", stating its ownership, users, experience, recommendations and whom to write to share or ask information about it.

A section on general purpose articles will be available to the teachers containing educational articles in electronic format. It should also gradually evolve into an electronic library, to be used in the same way as the applications.

The Cultural Center

Teachers and students will be encouraged to work on collaborative projects (i.e. story writing, scientific experiments), to engage in discussion lists and to build their own multimedia applications using general purpose authoring tools. These activities are organized and conducted in the "Cultural Center", which is a known place, normally next to a Plaza in small towns. Our Center contains an information space for every area of interest, as in some electronic bulletin boards. Users design their projects and use the e-mail system from inside this place.

For the 1992 experience, the Cultural Center was basically a bulletin board containing three discussion lists: Logo, Special Education (for students with special needs) and Sports.

The Post Office

The post office is presently a simple to use e-mail system. The letters are public documents inside each school. the Post is separated in four areas, for teachers, for students, for the newsletter editors and for the teachers in charge of the network at each school. Users can presently write and draw pictures on their letters, and will soon be able to send short sound messages. The main purpose of the Post Office is to establish initial contacts with people having similar interests in order to later continue a more structured communication inside the Cultural Center. It is also used for pen-pal initiatives and for informal messages, not particularly connected to any project.

Future Work

Future work will concentrate on several areas.

1) the user interface has to be revised, making a more lively Plaza. There are several opportunities to introduce meaningful multimedia capabilities to the Plaza, such as moving icons when e-mail arrives, sound or animations when newsletters are updated or a new item has been incorporated to the museum. The Plaza is regarded as a first step toward an "educational town". For example, a health care center will be provided to learn about personal hygiene, eating habits, etc. This is of particular importance in low-income areas.

Also, the underlying communications architecture will have to be improved, including more efficient ways of connecting the schools and the ability to handle an expanding network.

2) a teacher training strategy will have to be implemented using mainly short in-site workshops and the network for follow-ups. Although the software is very simple to learn, many teachers are frightened anyway, particularly if they have been exposed to older command driven interfaces. Most of the training will be devoted to methodological issues, such as how to prepare, conduct and evaluate an experience with students using computers.

3) the place for the computer at the schools must be carefully chosen. It seems that the library and the teacher rooms are strong candidates for extracurricular uses.

4) Much effort will be devoted toward designing new collaborative projects for teachers and students with reference to their particular social and cultural interests.

Conclusions

A national project on computers and education in Chile has been presented. The project has already started and will gradually expand until it reaches at least 100 nodes.

The project's assumptions that an attractive, relevant and simple to use software platform encourages the meaningful use of computers for stimulating a wide range of educational, social and professional activities for teachers and students in low income schools, are being validated.

The software design itself is being validated. The software has several control mechanism to record and find out the way students and teachers use the software and to measure the amount and type of traffic in the network.

Eventually, the project's final objective is to make public schools a more effective and attractive place for teachers and students regardless of their social, cultural or geographical situation.

References

Ambron,S.,&Hooper,K. (eds.) (1990). Learning with Interactive Multimedia. Developing and Using Multimedia Tools in Education. *Microsoft Press*, ISBN 1-55615-282-5.

Arancibia,V.,& Roa,M. (1986). Interacción profesor-alumno: Observaciones en sala de clases en escuelas básicas en sectores de extrema pobreza. in M Chadwick y I. Tarky (Eds.) *Dificultades en el Aprendizaje de las Matemáticas*. Editorial Universitaria, Santiago de Chile..

Bravo,C.,& Montenegro,H. (1977). Educación, Niñez y Pobreza. *Editorial Nueva Universidad*, Stgo, Chile.

de la Puente,R.,& Araya,D. (1991). Estudio sobre equipamiento computacional y su uso en los establecimientos educacionales de la región metropolitana. *Ministerio de Educación de Chile*, Depto de Planificación .

Fisher,K.M.,& Lipson,J.I. (1984). What Computers Can Do in Schools: Now and the Immediate Future. Petreson, D. (Ed.) "*Intelligent Schoolhouse*". Reston Publishing Company, Inc. Reston, Virginia U.S.A. ISBN 0-8359-3107-2.

Gillete,J.E. (1992). The New Language of Images: Observations on Computer Multimedia Form *Multimedia Review* , 3 (2), 20-29.

Hirschbuhl,J.J.,& Konet,R.M. (eds.) (1990). *Computers in Education*, Fourth Edition. The Duskin Publishing Group. ISBN 0-87967-836-4.

McAnge,T.R., Harrington,M.,& Pierson,M.E. (1990). A Survey of Educational Computer Networks. *Electronic Publication on the Internet.*

Moffat,A. (1972). Psicoterapia del Oprimido. *Editorial ECRO*. Buenos Aires, Argentina.

Muñoz,C. (ed.) (1988). Calidad, equidad y eficiencia de la educación primaria: estado actual de las investigaciones realizadas en américa latina. *Centro de Investigación y Desarrollo de la Educación*, Santiago de Chile. ISBN 968-7165-24-3.

Nelson,W.A.,& Palumbo,D.B. (1992). Learning, Instruction and Hypermedia. *Journal of Educational Multimedia and Hypermedia*. 1, (3), 287-299.

O.T.A. Report (1989). Power On! New Tools for Teaching and Learning. Superintendent of Documents, *Government Printing Office*, Washington, D.C. 20402-9325.

Park,O. (1991). Hypermedia: Functional Features and Research Issues. *Educational Technology*, 8, 24-31.

Ruth, S.,& Ronkin,R. Aiming For The Elusive Payoff of User Networks: An NGO Perspective. *Annual meeting of the International Society for the Systems Sciences*, Denver, Colorado, July 1992. VITA Distribution Service 24 Aug 1992. Size, 28.0 kB. Filename: ngo_perspective.txt

Sanchez,J. (1991). Educational Computing in Chile: Trends, Issues and Recommendations. *Educational Technology*, (3), 54-56.

Interactive Multimedia in a Distance Education Milieu

LOUISE C. HEWITT
Department of Computer Science
Illinois Institute of Technology, Chicago, Il 60616, USA

Traditional approaches to distance education include viewing instructional programs on television via videotapes, microwave, satellite, or on cable networks, accessing databases of instructional materials, or using electronic bulletin boards. Many programs on these platforms are of high quality, yet, each of these approaches assumes a passive learning environment.

A non-traditional approach to distance education assumes at a minimum, either the instructor, or one or more students, are situated at disparate locations, and are participating in the live class. This includes instruction platforms using cable, microwave, satellite, T-1/T-3 lines, fiber optics, ISDN, and personal computer-based teleconferencing systems when they are used in on-line, real-time education.

The incorporation of multimedia on a variety of distance education platforms became the objective of the Interactive Virtual Campus & Media Laboratory at Illinois Institute of Technology (IIT).

Background

The IIT Center, the midwest's largest private center of applied research in science and engineering, consists of the Illinois Institute of Technology and the IIT Research Institute. IIT is a private, independent, co-educational university comprising six colleges and schools: the Armour College of Engineering and Science, The College of Architecture and City Planning, the Institute of Design, the Stuart School of Business Administration, the Lewis College of Letters and the Chicago - Kent Law School.

Illinois Institute of Technology's ITFS (Instructional Television Fixed System) system, IITV (Interactive Instructional Television) was established in 1976 when the first computer science course was transmitted by microwave to the Sears Tower and then out to several corporate sites. The signal was transmitted from the IIT campus to a microwave dish on the Sears Tower, and from the Tower to a dish on the roof of each remote locations. These locations normally are within a 25 mile radius of the broadcast location. However, due to the relatively flat landscape of Chicago (A definite plus for distance education!), the signal reaches about 50 miles.

Microwave is a line-of-site communication system. Each remote location to participate in IITV courses must be in the line of site of the dish on Sears Tower. As it is currently the tallest building in the area, if not the world, this encompasses quite an area.

Today, IITV reaches over 50 corporate sites and 7 public sites. Locations such as Argonne National Laboratory, Fermi National Laboratory, Motorola, Zenith, Tellabs, among others, have employees enrolled in regular IIT credit-bearing courses. Students leave the work floor just before the class begins, and go to a TV room at the work site, equipped with a television monitor and a phone. Tuning to the correct channel brings the course audially and visually to the remote site, at the same time that it is being presented live at one of the IIT campuses. As questions arise, they can call the broadcast room.

At the back of each broadcast room is a red light which indicates an incoming call. When convenient, the instructor indicates that the caller should proceed and the call is piped into the room so that all of the students can hear the question, and the response. This talk-back system approximates a raised-hand, in theory.

Currently, 8 microwave channels are supported simultaneously during the day and evening with courses originating at IIT's main campus in Chicago and at its west suburban Rice Campus in Wheaton, Illinois.

Hundreds of students have completed degree programs at IIT without ever having to come to the main campus. The advantages of the microwave system for the student include:

. Minimal time away from the work floor
. No additional travel time or expense
. Potential for corporate sponsorship in tuition

and for the institution of learning:

. Additional student tuition revenue
. Publicity and exposure in the corporate site

IIT is also a charter member of NTU, the National Technological University, which uses satellite technology for the delivery of courses. IIT both downlinks courses from sites around the world, as well as uplinks some of its own courses.

Microwave and satellite programs are similar in the mode of delivery. Instructors are equipped with a microphone, a blackboard or white board on which to write. One or more cameras focus on the instructor or the board, appropriately. In some cases, an overhead camera may be available for transmitting images on paper or 3-dimensional displays such as chemistry demonstrations.

The extent of interaction between instructor and student is naturally limited. Call-in questions are the exception, rather than the rule. Students do not want to take the time away from taking notes to move to the phone, dial the classroom number and wait for the instructor to note the phone light and signal for the caller to proceed. If this is done once during the lecture or class, rarely will the remote student call in again, if they have another question.

This, of course, does not discount E-mail systems for student help. Rather it attests to the lack of support for the known advantage of immediate feedback.

Perhaps even more importantly, the quality and composition of the signal are at the explicit direction of the camera-person. If that person is not paying close attention, you may be writing on one board while the camera is focused on another. Moving back and forth between boards can be hazardous to your health unless the panning is done slowly, and the microphone is lavaliere style.

If several boards are used, those at the extremes are often seen on an angle and reading is difficult. Without experienced instructors, and top notch camera-persons, the delivery may be only fair or even poor.

Since the Spring semester 1992, IIT has delivered courses to employees at the IIT Research Institute in Annapolis, Maryland, using a T-1 (leased, dedicated phone) line. This line supports two-way video and two-way audio during the entire class. The instructor views the students at Annapolis on a television monitor at the same time that the students at Annapolis are viewing him or her.

As part of a regular microwaved course, the students at Annapolis do have a distinct advantage. They are able to interrupt, graciously, and ask questions without having to stop taking notes and dialing the broadcast room. The number of questions are asked by these students during a normal class period ranges from 10 to 20 times that by microwaved, call-in systems.

Clearly, the availability and ease of use of the T-1 system promotes a higher level of interaction between instructor and student than is possible with microwave and satellite systems, as is currently deployed.

Rationale for Research

Instructors using the IITV microwave system were forced to function with part of their class relatively silent. They were prevented from utilizing graphic and video images effectively in the distance education setting. The incorporation of multimedia products into distance education classes, and the ability to interact with students at a much higher level than was currently supported, drove this research.

Students of the 90s are brought up in an interactive, multimedia environment. They are accustomed to participating in multi-way phone conversations, to viewing high quality color graphics in newspapers and magazines, and spend considerable time watching videos.

In 1988, technology was available for interactive, multiway teleconferencing, graphics and video production using the personal computer. Integration of each of these systems into the available microwave system was the initial goal.

Sound

The first step in testing the extent to which a higher level of interaction would benefit remote students was based on interactive audio. In this context, it referred to the ability of the instructor and the students to be able to speak at any time during the class using a speakerphone system. To effect this, a conference call was established for each meeting of one of the author's classes using the AT&T Alliance teleconferencing system. Remote locations were connected to the classroom phone in a conference call, and the call was continued during the entire class session.

The results were extremely positive. Using the old call-in system, at the most, one telephone call was received during each class session, by all remote locations combined. Using the teleconferencing system, where all sites were connected for the entire class, remote students asked many questions during each class session. At times, they asked other students to elaborate on questions they raised, or projects that they had described.

Clearly, the ability to ask questions easily, without interruption of notetaking, was a considerable advantage to the students. Nor did they did not feel stigmatized against asking multiple questions. The interactive audio system also reduced the number of telephone calls to the instructor, after class, where answers were often repeated many times as different students called with the same question.

If the increase in participation based on audio was so successful, the addition of graphics, video images, and annotation of these images should be more successful. Add to this the capability for annotations and storage of data, and the result might be outstanding.

Graphic Images

Many of the instructors at university use computers in the classroom. The computers may be used to demonstrate topics in computer science, such as using a compiler, or as a tool to demonstrate software in support of other disciplines. Chemistry simulations, math solver techniques, computer-aided design, are but a few such applications.

The transmission of the computer screen image to remote television monitors poses significant problems. Television monitors are analog systems and computers are digital. If that was not significant enough, where you are located in the world and where your audience is located, also poses significant problems.

There are three major worldwide television standards, NTSC, PAL and SECAM. NTSC is the standard used in the United States, PAL is used in England, much of Europe and Australia, and SECAM is used in France, and other countries.

In 1948 the National Television Standards Committee (NTSC) met to create a national television standard. It took until 1953 for the competing companies, broadcasters, and the FCC to arrive at a uniform standard. While technology has improved, this standard is still in use today. In fact, other than adopting a color TV standard allowing black-and-white TV receivers to work with color signals, little has changed.

NTSC uses a scan rate of 30 frames per second (fps), which is a function of the available frequency of AC power (60 cycles per second), resulting in 525 scan lines. (PAL, for example, is based on a 'scanning frequency' of 25 fps on a picture of 625 lines.) NTSC is an analog signal, and is interlaced which means that the odd numbered lines are scanned first and then the even numbered lines are scanned. If you have seen computer monitors on television, you may have noticed a rolling motion down the screen. This is the difference in scan rates. To display computer images on a television, the computer digital signal must be converted to an NTSC signal. Keep in mind that this is about 10 times the ISA bus speed of the PC. Prior to the availability of adapters to provide VGA - to - NTSC, signals, the easiest way to include computer screens in a distance education format was to focus the camera on the screen. However, as a result of the difference between the scan rate on the computer screen with that of the television, a roll was apparent. This may not be significant if the image was to be held on screen for a short time, but for any length of time, it was very tiring on the viewer.

To solve this problem, a survey of the industry (1989 - 91) was done and dozens of video systems tested. Some of the problems encountered included shimmer between certain colors when juxtaposed, and elimination of part of the computer image. A number of VGA-NTSC adapters are currently available and offer a simple solution. Many of these add-in cards use a process called Gen-locking, or synchronizing the 31.5-kHz VGA signal with the 15.735-kHz NTSC signal.

The Magni Producer was selected for transmission of computer graphic images. It provided the means by which the signal could be interfaced with the video switcher in the broadcast studio. The composite output from the Magni is fed directly into the switcher and that signal is transmitted over the microwave. The image was of high quality with no roll, colors remained true and no loss in image was encountered.

The Magni includes an external control unit with cut and fade buttons and video/graphic slidebar selector. Once initialized, it unit could be used without resetting.

The ability to present high quality images without roll or flicker significantly improved the quality of our classes. Students on-site were able to view images on several large screen television monitors in the classroom, while remote students were able to view the image on their regular monitor.

Additional responsibilities for the instructor were minor. In fact, the ability to use computer screens in instruction saved time in preparation and in describing computer sessions. Instructors in many disciplines are currently taking advantage of this system in courses ranging from CAD to Business.

Student feedback is very positive on the inclusion of computer based graphics in instruction and the high quality of the presentation material capable with the system.

Video Images

The ability to include video images was the next step in adding additional media to the platform. A Targa Video Imaging card was added to the configuration. The use of the Magni interface was extended to take advantage of video images. These included camera-captured images, VCR extracted images, and other sources. A slidebar on the Magni unit switches between graphic and video settings.

Extended Interaction - Handwritten and Typed Annotations

The interaction in the delivery of education is closely related to learning. The more interesting and dynamic our presentations, the more stimulated the students, the more learning will take place. The addition of graphic and video materials into the classroom was a giant leap into the use of multimedia, but left us wanting more. Information is dynamic rather than static, changing as the subject matter changes, expanding as new information becomes available. It was not practical to switch from the computer-generated image to the blackboard and back again, repeatedly. The need to annotate directly on the computer image was apparent.

Simply typing at the keyboard would provide some measure of dynamic input, but the quality of the typed characters was too low. They were too small to be seen clearly both for the on-site and the remote student.

A number of annotation systems were researched with wide differences in price range. Some temporarily froze the screen, supported annotations, and then allowed you to return to the running application. This proved cumbersome and timecomsuming. The system chosen was the Sony Video Sketch Titler. This low priced, easy to use, and readily available tool, interfaces between the Magni output and the video input. A stylus permits annotations in many colors, widths, and filled and unfilled for highlighting.

The Video Sketch Titler also provides two simultaneous displays, so you can switch between one application screen and another screen. A built-in titler provides the capability of adding text to images. Built-in graphic images may be made to appear and wipe away in several ways, all adding to a dynamic, colorful presentation.

Instructors have found this an easy to use, intuitive system, for the colorful on-line annotation of images. The system is adding dynamic capabilities to the on-site classroom, and to the distance education delivery systems.

Storage and Retrieval in a Distance Education Setting

The ability to display complex images and concepts in a visual manner became both a tremendous advantage as well as a potential area of concern. One of the key advantages of using multimedia in the class was to be able to remove the necessity for explaining how a image or action would appear, by causing it to appear. The inclusion of images on-line and in real-time, provided the instructor with considerably more time to discuss the concepts and techniques in the application, rather than drawing the image. The storage and retrieval of these images, and even the annotations was necessary.

The ability to incorporate graphic and video images into classroom presentations necessitated a survey of available products purporting to support video capture and display. The products chosen must also be capable of supporting all of the graphic features that we have previously discussed.

Several systems were tested, and one selected as the most cost effective, and easy to use. Optel Communication Inc.'s Telewriter system seemed to provide the means by which:

. annotations could be drawn or typed

. graphic images could be displayed

. video images could be easily captured from a video camera, a camcorder or VCR via a video imaging card, and easily displayed

The Telewriter teleconferencing system supports the display of text, graphics and video images on one multisync monitor (using a switcher card) or on two monitors, one for graphics and one for video.

The software supports the storage and retrieval of graphic and video images in a vertical and horizontal manner. Storage of images is accomplished through the use of a video source such as a camera and the Targa imaging card. A single keystroke captures the video image and saves it either as a Targa video image or converts it to a .PCX graphic image. Databases of images (both graphic and video) as Telewriter pages may be selected and called-up at will. Directories may be accessed in any order, and on-line annotations in four colors and two widths are supported.

Of major significance is the capability to save annotations, both typed and drawn, either on the base image or as a separate file. These can then be viewed at a later time.

To support the display of graphic images and video images on a display, and to support the printing of graphic and video images, and annotations, the Illinois Institute of Technology's IVC & Media Laboratory has authored software called WorkBook which will display and print .PCX, video and Telewriter pages automatically. Thus students could leave the class with a disk containing the images displayed in class and review them outside of class. Conversely, instructors could provide students workbooks on which handwritten notes could be added during the actual class presentation.

Using WorkBook, students will not have to take copious notes without really listening, teachers will not have to step through many packages to print graphic and video images and annotations. This opens the way for a higher level of classes, where active participation and discussion are possible.

Distance Education Expanded

All of the systems discussed act as supplementary aids within a microwave (or satellite) system. For the distance education student beyond the microwave, for whom the individual or the corporation cannot support microwave equipment, for the homebound or institutionalized for whom advanced telecommunications systems are unavailable, a different, cost effective system was needed.

The IVC & Media Laboratory took advantage of the Telewriter's distance education - teleconferencing support by using the personal computer and the POTS (plain, old telephone system) as the vehicle for distance education.

The Telewriter system supports multiple (up to 99) sites in a conference over the standard telephone line. A special multipoint modem was used so that one phone line supported both audio and data. Graphics tablets at each site were optional, but useful. While text may be sent on-line, in real-time, most graphic and video images were transmitted prior to class. During the class, students and instructor interacted audially with a speakerphone, digitally using the keyboard, and graphically with a graphics tablet. As the instructor called-up an image on his computer, the same image was immediately and automatically called-up on the student's monitor. Annotations drawn on the instructor's tablet (or the students') or typed at the keyboard were seen by all participants. Permanent storage of all data, text, graphics, video images and annotations was via the hard drive on the computer.

Preparation of class materials in advance was important so the images could be sent to each remote site prior to class. The easiest system was with the use of a single Bernoulli disk (90M) for class materials. When the instructor came to class, a personal computer - distance education workstation with a Bernoulli drive enabled the instructor to simply insert the disk for that class. An instructor development lab provided facilities for video and graphic development and storage.

The remote, distance education students participating using the personal computer teleconferencing system, were part of the regular class delivery system. Training for the remote students prepared them in the use of the hardware, and the techniques of participating in a remote classroom.

Current Applications

Approximately half of the computer science faculty are using some or all of the multimedia systems in their classes, those based on microwave, satellite, or leased phone lines. Several engineering faculty and business faculty are in process of integrating these systems into their teaching.

Some of the remote sites opt to receive the cable/microwave broadcast only. Others may add the audio component so that they are connected audially during the actual live broadcasts and can take advantage of verbal communication.

Still others opt for the complete, interactive teleconferencing environment. This includes a personal computer equipped with a graphics tablet, multipoint modem and teleconferencing software. In some cases it supplements the microwave system, satellite, or leased line system and in others it is used as a stand-alone educational delivery system for live university courses.

The broader the capabilities, the greater the number of faculty who will take advantage of multimedia systems, the more powerful and dynamic the instruction. It is our contention that students taking courses using these media will show an increase in attendance, a greater degree of in-class participation, and higher retention.

Expanded usage is being made outside of the university setting. As part of in-service education, IIT is offering courses to teachers in the Chicago public schools, over a

ERIC

microwave to cable system along with the Telewriter teleconferencing system. Every school in the city of Chicago is cable-ready. During the Spring 1993 semester, teachers in ten area schools will enroll in a university course, CS 460 (Fundamentals of Multimedia). This course introduces multimedia in theory and practice for use in the classroom. The course originates at the university and is transmitted to the cable company via microwave which is then converted for output via the cable system. With the backbone of the Telewriter system, each teacher is a true interactive participant in this multimedia arena.

Future

The last remaining step is the inclusion of high quality motion video at a low, end-user cost. In the immediate future, and for considerable time to come, analog will be the international standard just as the VCR is the standard delivery platform for video currently. NTSC, PAL, and SECAM broadcast channels represent the ultimate wide-area network. Combine these with Cable TV, and the analog video market explodes. In the long-run, digital video will probably be as commonplace as analog video is today. The main reason is the flexibility with the digital signal. This includes computer-controlled data, portability, storage and so on. NTU changing to digital during the past year is one such indication.

HDTV (High Definition Television) is another such indication. Four of the five proposed HDTV systems are all digital. Images are processed as well as transmitted digitally. Images will be sharper (doubling of the scanning lines), there will be new image-processing techniques, and a host of new applications will emerge. The FCC plans to choose an HDTV standard before the end of 1993, and HDTV receivers and broadcasting will probably be available by 1995. Coupled with advances in digital communication networks, the capability of participating high quality, dynamic and interactive instruction will be as easy as answering your phone.

ERIC

Creating Excitement and Motivation in Engineering Design: Developing and Evaluating Student Participatory Experience in Multimedia Case Studies

SHERRY HSI
Graduate Group in Science and Mathematics Education
University of California at Berkeley
Berkeley, CA 94720, USA

ALICE M. AGOGINO
Department of Mechanical Engineering
University of California at Berkeley
Berkeley, CA 94720, USA

Background and Motivation

As a part of a national effort to improve engineering education in a collaborative effort among eight universities that form the Synthesis Coalition (Cal Poly San Luis Obispo, Cornell, Hampton, Iowa State, Southern, Stanford, Tuskegee and University of California at Berkeley), an experimental investigation is being performed in developing a more integrated curriculum using educational multimedia. In the traditional curriculum, engineering students who try to integrate knowledge and concepts may find learning a frustrating and difficult experience, especially in their first two years of college. Most of the courses involve learning abstract, declarative knowledge, analysis and problem solving with little opportunity for synthesis, creativity, self-directed investigation, and exposure to the field of engineering (Agogino, Sheppard & Oladipupo, 1992). Although design courses aimed at integrating knowledge are typically offered at the senior level, many students lose interest and motivation in engineering before they even find out about the engineering discipline.

Our project is aimed at teaching design principles, as well as creating excitement and motivation in engineering by providing 1) illustrated examples of inter- and multidisciplinary designs using multimedia case studies, and 2) opportunities for students to extend these cases. Based on the case study method of teaching engineering design (Sansalone, 1990; Steidel, 1973), we used multimedia to capture multidisciplinary aspects of the engineering design process drawing cases from successful student competitions and exemplary industrial practices. In addition to teaching engineering design, cases are designed to contain historical connections, social implications, design rationale, market-driven technologies, and profiles of engineering role models and design participants.

Multimedia/hypermedia provides integration of hypertext with digital video clips of interviews with industrial designers, engineers and managers. Tooling and manufacturing processes are also shown along with relevant texts, scanned still images, CAD drawings, and audio voice tracks. By creating a collection of instructional multimedia cases, we hypothesize students will gain a better understanding, perspective and appreciation for the engineering design process and the discipline. Also, early experiments indicate that the activity of developing their own case increases their interest in engineering and encourages their participation in engineering design activities.

Description of Multimedia Cases

Several case studies are currently under development on the Macintosh and IBM platforms (Hypercard™ & Toolbook™ authoring tools). Two cases designed for different age audiences (lower

ERIC

and upper division students) and different educational objectives will be presented here which include the Human Powered Vehicle, and the Proprinter cases.

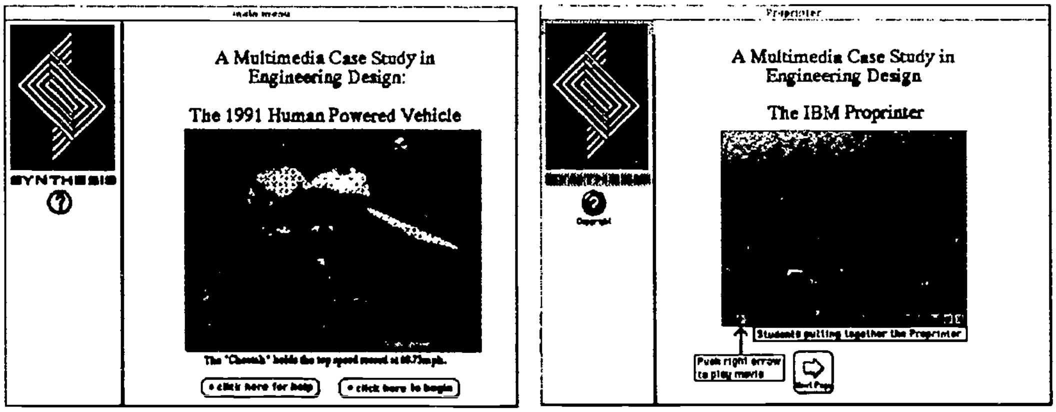


Figure 1. The HPV and Proprinter Cases

The Human Powered Vehicle : A Case Study Prototype

The Human Powered Vehicle (HPV) was chosen as an initial prototype to test the concept of a multimedia case study. The HPV Project is an extracurricular design activity on the U.C Berkeley campus that attracts engineering and science students to participate in a year long competition to build an aerodynamic vehicle. The HPV was ideal in that it covered many engineering disciplines and complex problems typical in concurrent design and engineering decision-making. The topic of a competitive race and vehicle design competition was suitable for generating interest and excitement to engineering students. It also provided an engineering project that was also accessible to students who wanted to participate in its design and construction.

Instructional objectives of the HPV case include conveying the multidisciplinary aspects of design, showing the importance for fundamental math, engineering analysis and testing. Both the process of constructing a case collaboratively by engineering students during the construction of the vehicle, and the outcome of a multimedia case served as curricular activities and materials. The case contains video clips of wind tunnel testing, road testing, vehicle construction, and clips of the actual race. Hypermedia is used to link relevant equations, drawing and graphs. By providing ties between theory, application, social-historical implications and global impact to their studies in engineering early on, we hope to provide ties between physical examples of engineering to theory and analysis learned in class, as well as change attitudes and encourage more students, especially women and underrepresented minorities to pursue engineering as a career.

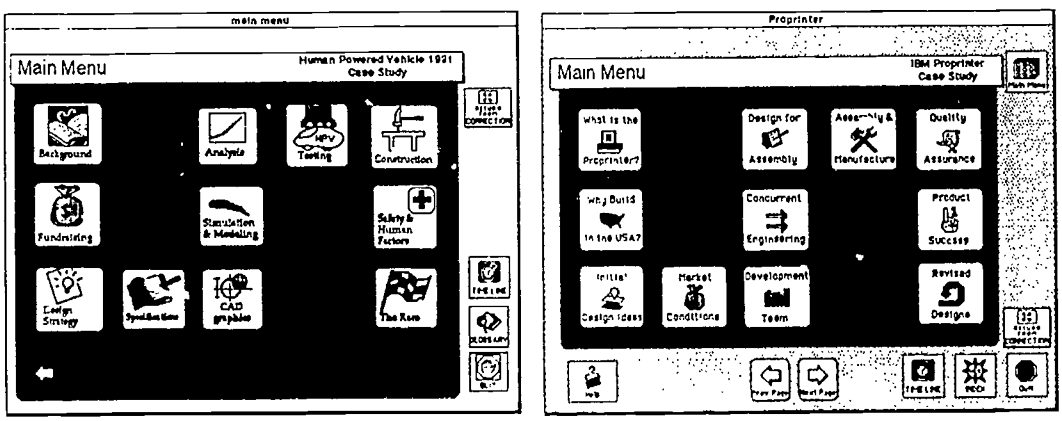


Figure 2. Design life cycle path for the HPV and Proprinter Case

The HPV case study supplies the user with two main navigational directions, one based on the design life cycle, and the other based on a chronological perspective (Figure 2). Each of the navigational approaches serves to structure the content, but also allow students the flexibility tochoose their own path of browsing and learning the material. To view materials non-linearly and make broader leaps, the "See Connections" button opens a browser that allows users to follow suggested paths locally within the stack, or link remotely to other relevant on-line materials such as other cases, theory from textbooks, and future video/image databases.

To engage the student in active thinking, interactive pop-up "*think*" questions are provided throughout the stack to stimulate reflection and critical thinking. These include both open-ended questions as well as questions where the answers lie within the multimedia case.

To provide role models in engineering, especially highlighting women and minorities, digitized portraits and voices are also attached to *think* questions. (In the HPV case, the faces are taken from U.C. Berkeley professors, local heroes, community figures and others, but the portraits can be customized for each locale.) The case study also highlights role models in the Who's Who section where faces, interests and contributions of the student designers are included.

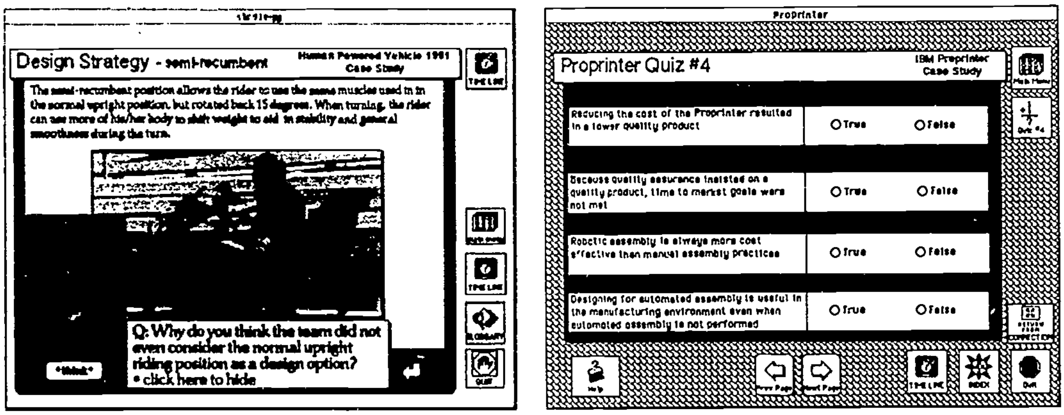


Figure 3. Example *Think* question and on-line quiz

The Proprinter Case: An industrial case study

A case study of the Proprinter, a dot matrix printer designed for automated assembly, is built upon the same life cycle design structure as the HPV (Figure 2). A linear presentation also allowed the option to pursue non-linear links to other parts of the case. Unlike the HPV case, the Proprinter case is targeted for upper division students and possesses different educational objectives. These include specifically teaching students about industrial practices of automated assembly, multifunctional teams, design trade-offs, and concepts in engineering design such as concurrent engineering and design for manufacturability. The Proprinter case contains on-line quizzes to encourage self-monitoring and checking. If students fail to answer the questions correctly, the system prompts them to review the related material, and links them to the appropriate section. Also, the Proprinter takes advantage of computer animation to describe and disclose the manufacturing processes and robotic assembly.

The Proprinter was also designed to be motivational as it links current technology to historical figures through the "See Connections" button. In this example, the concept of quality is linked to Elijah McCoy, the son of an escaped slave (Figure 4).

Using Multimedia Cases in an Instructional Setting

Multimedia cases can be used in a variety of instructional settings. At Berkeley, faculty use case studies to support standard lecture material using an overhead project panel to display relevant portions of the case to help illustrate more abstract concepts. (An index allows instructors to find topics quickly). Multimedia cases are also made available during computer laboratory hours for students to explore in more depth on their own. We also introduce case studies to pre-college and incoming students to spark interest and bring relevance to their own studies or projects in engineering. As more cases are developed, we plan to integrate courseware into the existing case studies course in engineering design which currently uses textbook cases as the basis for discussion.

Finally, multimedia cases are also used as exemplary models to fuel and motivate students to construct their own understanding through case study development. Through research and exploration, they are forced to reflect on explanations of concepts, and create multipie representations of engineering processes when they need to convey their ideas to other students and instructors. Students initially participate as a viewer of cases, and eventually take part as a designer-builder-explainer of their own multimedia case. Recently, an experimental course in multimedia case studies has been offered at U.C. Berkeley for first and second year engineering students which resulted in a section on the "History of the Bicycle". Students provided links to the HPV case, as well as extended an existing case on "Bicycle Dissection" which breaks down the design of a bicycle into its functional components.

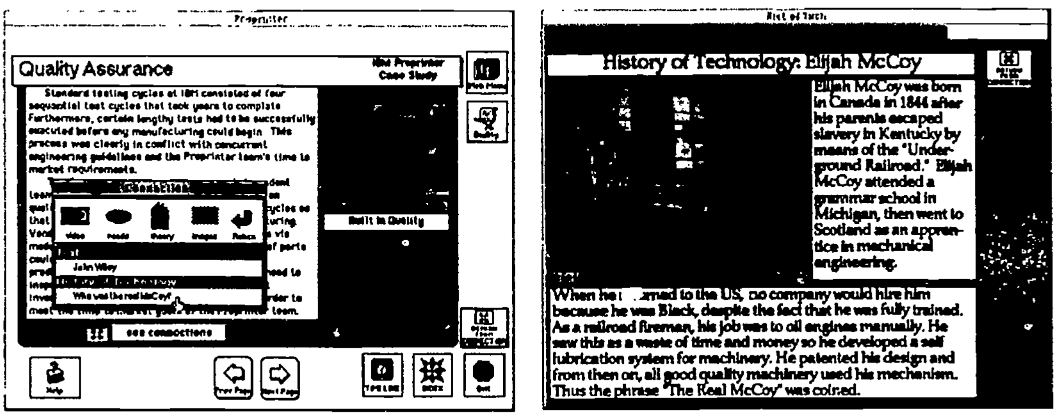


Figure 4. "See Connections" showing a historical link

Method of Evaluation & Assessment

The evaluation of educational multimedia is still in its infancy as the development of multimedia tools and the assessment of the impact on teaching and learning is still evolving. More recently, researchers are designing evaluation methods for curriculum innovation (Grabowski, 1992; Mandinach & Cline, 1990). We present multiple methods of assessment used to gather information about multimedia cases as curricular materials including interviews, discussions, surveys and an on-line data collection program. Our model for evaluation was one of trial and refinement where versions of the multimedia cases were rapidly tested, redesigned, and updated based on the evaluation results. This trial and refinement model was appropriate, not only because the cases were easy to modify, but because new technological developments from the multimedia/computer industry necessitated continuous integration of new technologies to keep cases updated from an implementation standpoint.

Several populations were selected for evaluation of designated cases which included students from an introductory engineering design and graphics course at the University of California at Berkeley, upper division students taking a computer-aided design course, summer students enrolled in introductory programming course, and junior high school students from a minority outreach program. Several faculty members also evaluated the multimedia cases.

Personal interviews and discussion

We performed initial evaluation through observational studies of students and faculty using the courseware. Comments were also collected via interviews with individual testers, from students taking the multimedia case studies course, and from group discussions including graduate students in education technology. Qualitative comments about user interface design, difficulty of use, value of content, and use of multimedia to illustrate concepts were collected, documented and fed back into the design of the multimedia cases.

Pre and Post Surveys

The purpose of the surveys was to collect two categories of information. The first kind of information related to the ability for cases to meet the educational objectives and pedagogical intent of each case, and the effectiveness of multimedia cases to deliver those concepts about engineering design. (In the Proprinter case, users were asked to answer open-ended design questions before and after looking at the case.) The second category of information pertained to the technical aspects of the courseware to aid in the improvement of the cases such as video quality, quantity of audio, ability to navigate, functional aspects of the user interface. Questions were designed to gain information about the quality of the multimedia elements, user interface, navigation and overall presentation of the multimedia case. The survey consisted of both multiple choice questions, as well as open-ended response questions.

The procedure for survey evaluation included first administering a 20 minute pre-survey, then instructing students to use the computer to review the multimedia case for 30 minutes. Finally, students were asked to take a 20 minute post-survey.

On-line Data Collection

Another source of evaluation data was collected by an on-line computer program. To study individuals interactions with the courseware, and for purposes of navigation assessment, a computer program was developed to collect the times and location where students chose to browse in the case study. This quantitative information gave us insight into what order they chose to view the material, which sections they chose to look at, and how long they spent on each screen or section. From this data, we inferred which parts of the courseware students found more interesting as well as which paths were easier to navigate along, and which paths were more difficult.

Results

In general, evaluation data suggest several improvements necessary in the case studies presented here, but overall the cases were found to be very successful in conveying principles of design, generating excitement about participation in engineering design activities, getting students to learn engineering concepts through participation of case studies development, and teaching current concepts found in industrial practice.

Results of Interviews and Discussions

Given thirty minutes to an hour to view the HPV case during evaluation, some students complained there was too much material to look at. Many of the undergraduate students wanted less technical detail and a more general overview of the topic. On the other hand, some students wanted more details about the design, and wanted to see links to development environments to modify the design of the human-powered vehicle in an interactive computer simulation.

When ask what part of the HPV case students liked the most, there were a variety of answers indicating that students found different parts of the case interesting. The on-line data collection supports this indicating there was always one section where students spent a disproportionate amount of time.

Students taking the experimental multimedia case studies course expressed positive feedback for the course and interest in continuing multimedia case development in the following semester.

On-line Collection Results

On-line data collected revealed that students chose to spend different amounts of time studying different sections. In the HPV case, some liked the who's who and safety issues, whereas others liked the sections on analysis, testing, the race, the timeline, and specifications of the vehicle. This variation indicated that students could find something in the case they found interesting related to design and engineering.

ERIC

In the Proprinter case, the quiz questions were found to encourage more careful reading of the material. More time was spent answering the first quiz (due in part because of the need to return to the text to get the answers). Also, data indicated students spent less time on subsequent quizzes as they progressed along in the multimedia case as a result of better self-monitoring.

On-line data collected regarding details of navigation, user-interface testing, and functional aspects of the HPV interface are discussed in more detail elsewhere (Hsi & Agogino, 1993).

Survey Results

Survey responses collected from both undergraduates and junior high school students indicated that older students were more critical of the quality of multimedia elements and had higher expectations of video and animations. The students demanded higher quality movies and videos that were "less choppy". (The Macintosh IIcx machines with 8-bit color were used during testing.) As a result, several video clips were redigitized to allow a higher quality images and smoother playback.

Students wanted more digitized voices of the HPV team members, and longer video clips where current clips were 30-60 seconds long. Students were annoyed at repeated videos, and as a result repetition was eliminated. Although Quicktime movies allow students to adjust the volume, students complained they could not adjust the volume of digitized voices easily. Also, a large computer laboratory using multimedia cases demanded use of earphones.

In the Proprinter surveys, student responses to design questions were rated from 1 to 3 (poor to good) based on their overall conceptual understanding of design and use of keywords. After viewing the case, their score increased from an average of 1.62 to 2.46. When asked to make suggestions to a redesign of a product made from multiple plastic and metal parts assembled with solder and screws, one students response went from "I don't know" to "eliminate screws and try to use less pieces" indicating the Proprinter case successfully delivered engineering design practices and concepts (Evans, 1992). Surveys also indicated students attitude towards design changed, considering it more as a team effort.

The *think* questions which were designed to promote reflection in the HPV case instead caused much frustration among the students and faculty. More than half of the students surveyed said they were not serious about answering the think questions, or abandoned answering them altogether. The think questions were found frustrating because many questions were open-ended and the answers could not be directly found in the case. This has prompted a redesign of the *think* questions to reduce the number of open-ended questions, and to include more questions directly relevant to the material in the case. Several faculty members have offered to provide and integrate relevant questions from their own courses to put into the multimedia case which could serve as interesting links from their lectures into the multimedia case. In our instructor's guide, we suggest that some *think* questions can be used to stimulate class discussions or be used as part of the problem set.

Although there were some complaints about various aspects of the implementation, almost all of the students after seeing the case study expressed enthusiasm for joining in the design of the HPV, and taking a course which provided these case studies as curricular materials or participating in the development of cases as a design activity. Their post-survey responses also indicated after viewing they found design to involve more multidisciplinary factors than previously thought. They learned that design also requires fund raising, team work, project planning, engineering analysis, and other factors such as safety considerations, marketing and cost.

Future Work

Cases on various toys designed by Mattel, Inc., are currently being developed emphasizing market-driven design concepts, manufacturing processes, design for assembly and safety concepts. Plans include porting cases onto a UNIX platform to allow for faster networking and links to remote databases and simulations. Plans to research, modify and improve current evaluation methods of educational multimedia are also underway. One challenge will be to evaluate the benefit of case studies as a scaffolded constructive learning activity.

Our ultimate goal is to distribute multimedia cases to many university campuses as well as introduce cases as a motivator for pre-college students initially on a CD-ROM format, and later on a networked courseware database. Contents of case studies will be tested for sensitivity to cultural,

ERIC

ethnic and gender differences to strengthen the increase pipeline of students coming into engineering.

Conclusions

Multimedia case studies have been presented as one approach to a multimedia curriculum for generating excitement in engineering. Multimedia also provides a mechanism for assimilating engineering design concepts in an illustrated context. Evaluating the educational objectives was achieved through the use of multiple methods of assessment where the results were used to rapidly modify, retest, and refine the curricular materials.

Learning engineering still comes from practice and hands-on experience rather than simply using interactive multimedia. However, the courseware can be supplemented with hands-on experiences in lab settings. For instance, the Proprinter case was used in conjunction with students participating in the dissection and re-assembly of an actual printer. Stanford University is using the "Bicycle Dissection" case with bike dissection activities (Agogino, Sheppard & Oladipupo, 1992). Providing these examples of exemplary design in the classroom, coupled with hands-on experiences, we can raise curiosity and interest in the discipline, and ultimately help provide a perspective of engineering that is more integrated and accessible to students engineering early on.

References

[1] Agogino, A. M., Sheppard, S. and Oladipupo, A., (1992), "Making Connections to Engineering During the First Two Years", Proceedings of the Frontiers in Education Conference (Nashville, Tennessee Nov. 12, 1992), IEEE.

[2] Grabowski, Barbara L. (1992)"Evaluating Interactive Multimedia for Learner Engagement" Annual Meeting of the American Educational Research Association

[3] Hsi, S. and Agogino, A.M. (1993) "Navigational Issues in Multimedia Case Studies of Engineering Design" to appear in Proceedings of the Fifth International Conference on Human-Computer Interaction, Orlando, FL August 8-13, 1993,

[4] Mandinach, E. M. and Cline, H.F. (1991) "The Impact of Simulations and Modeling on Teaching and Learning Activities in the Classroom", Annual Meeting of the American Educational Research Association

[5] Sansalone, M., (1990) "Teaching Structural Concepts Through Case Studies and Competitions", Engineering Education, May/June, pp.472-475

[6] Steidel, R. F. and Pefley, R.K. (1973) Engineering Case Method Experimentation at the University of California, Berkeley.

Acknowledgements

The Interdisciplinary, Multimedia Case Studies in Engineering Design were developed by faculty, graduate students, and undergraduate students at the University of California at Berkeley in collaboration with our partners in Synthesis: an NSF Engineering Education Coalition. The authors are grateful for the support of the National Science Foundation and our industrial partners: Apple Computer, Inc.; IBM Corporation; John Wiley and Sons, Inc.; Lexmark International, Inc.; and Mattel Toys, Inc. Special thanks goes to case study developers: Jay Evans, Anil Varma, Nagaraj Srinivasan, Punit Jain, Armineh Menzigian, Kirk Edwards, Bill Wood, Steve Bradley, Mitchell Suarez and Robert Stanard. We are indebted to our courseware studio specialist, Robert Lettieri.

Simulation Based ICAI with Multimedia

Masahiro Inui, Satoru Myojin, Katsuya Oba
OGIS Research Institute Co., Ltd.
2-95, 3 Cyome Chiyozaki Nishiku Osaka, 550 Japan

Using simulation, beyond the capacity of human instructors, is one of the best methods of computer-aided instruction. Recently computers have become powerful enough to execute complex simulations and make pedagogical plans. The simulation models used in training may differ from those for research or analysis because the abstraction of domain knowledge is important in training and detailed analysis is not needed. The qualitative model provides one concept.

The features of simulation based training systems are:
(1) Learners have various experiences of the objective world in imaginary environments and gradually establish mental models inductively. (one type of model instruction)
(2) Coursewares like simulation games do not bore the learners.
(3) Learners can grasp concepts easily through visualization without the need for complex natural language understanding.

The systems are being used in fields such as plant operations, troubleshooting, physics, chemistry, and language education. Here are some of the desired conditions of simulation models.
(1) quick responses to learner operations
(2) simulation models well merged into courseware
(3) each step of the situation easily grasped
(4) simulations done from the expert point of view
(5) excellent authoring system (builders) provided

OGIS Research Institute has developed two intelligent tutoring systems (ITSs) in cooperation with Osaka Gas: PCTS (Process Control Training System) based on physical models and CETS (Conversational English Training System) based on linguistic models. This paper describes the basic concept of ITSs for plant operation and conversational English and their user interface.

The Features of PCTS

The importance of CAI systems in industrial environments has become more apparent largely because of an increasing recognition of the complexity of the system operations to be mastered. We believe CAI systems have a more important role to play in such environments than in schools, which need the human touch between teachers and students. The operator training in Osaka Gas is classified into two types: central and local training. Central training is at training centers with a number of trainees using real equipment or large simulators. Local training is carried out at each factory or plant. PCTS is being used for the local training without any instructor.

The Structure of PCTS

1. Training Methods

Fig. 1 shows the basic concept of the PCTS architecture. The features inherent in the training methods used in GTS (Generic Tutoring System) are described below.

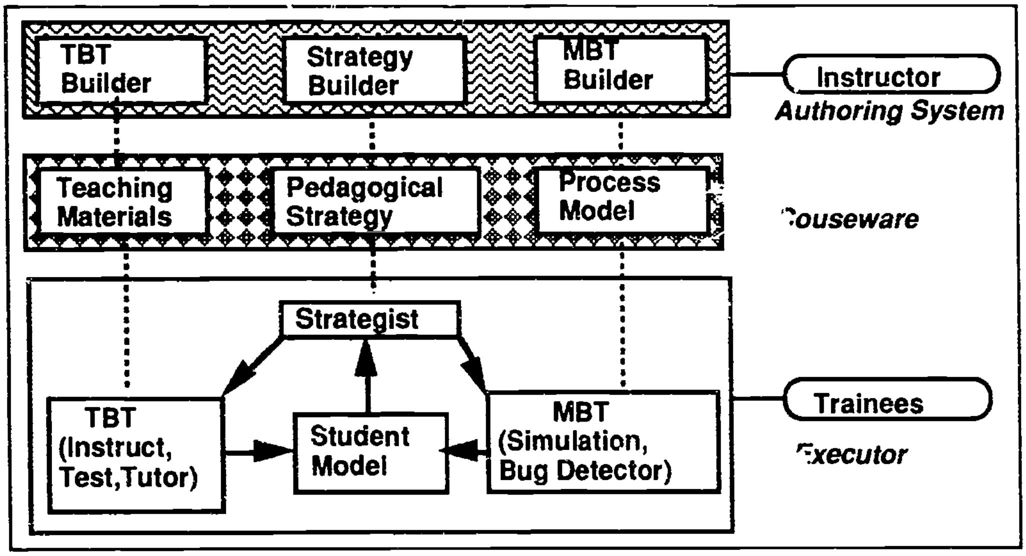


Fig. 1 The PCTS Architecture

(1) Training according to the trainee's level of understanding

Trainees take a pretest to determine their level of understanding before training, and are initially placed at different levels of training accordingly.

(2) Integration of TBT (Text-Based Tutor) and MBT (Model-Based Tutor)

The weighting towards textual lessons with supportive graphics and simple modeling allows common concepts to be initially presented in a direct instructional mode. As trainees grow in sophistication with regard to the operations, our training system moves toward model-based lessons. Current educational theories and research indicate that learning to apply knowledge and the transfer of knowledge to new situations require more hands-on and inductive presentation techniques, namely, learning by doing.

(3) Training according to progress

Each course consists of several lessons. each of which is composed of four modes: Instruct (lecture), Test, Tutor (reinstruct), and Retest. These four modes are ordinarily given in that order.

In TBT, Test has been categorized by Barrett into literal, inferential, applicative, and evaluative tests. If Test is not passed, Tutor is started. Tutor is divided into Repeat, Keyword, Hint, and Help modes in order to give the most appropriate advice according to the analysis based on the type and results of the Test.

2. Bug Detection and Remedies

The Bug Detector (BD), or trainee error detector, is used in the MBT to evaluate the trainee's performance in its best mode. It analyzes the trainee's behavior and posts assertions into the student model. Nine kinds of bugs have been identified based on plant operations.

We classified bugs into two categories, major and minor. If an operator causes major bugs, PCTS will reinitialize the process. If an operator causes minor bugs, operation can continue. If an operator causes both major and minor bugs at the same time, PCTS sends a message for the major level only. When a trainee finishes an exercise, one of three different messages appears, depending on the total number of misoperations. PCTS gives a hint the first time a bug occurs, and gives help the second time.

3. The User Interface and the System Environment

(1) The User Interface
Some trainees are not used to using computers, so operating the program should be easy and the outputs should be easily understood. The standard trainee input is by mouse. The output of the training system consists of views and statements assigned by X-Window. Graphics and static picture images are used for the views. Statements are represented with text supported by voice synthesis. Using these modes of interface makes training more effective.
(2) The System Environment
In PDTS, that is. our first training system for our plant operations, independent knowledge representations such as frame, rule, and OOP are used. The concept of fitting a knowledge representation to a problem has worked fairly well, but suffers from the relatively large number of independent representations needed and inconsistencies in the paradigms among them. Thus a unified architecture usinp an entirely object-oriented paradigm has been designed. It is implemented with C++ whi ·: is OOP of C language.
The PCTS is run on NEWS (an engineering workstation made by Sony) with a high resolution color bit-mapped display and a Japanese voice synthesizer (made by NTT) which is the same as the delivery hardware platform.

Hierarchical Layers for Operator Training

The PCTS system consists of three layers, namely, Text Based Tutor, Mini Simulators (MBT), and Exercises (MBT), as well as simulation models. The hierarchical layers for simulation-based ITS have been designed on the basis of the experience of PCTS, PDTS and hierarchical layers made by R. Gagne.

1. The Hierarchical Layers made by R. Gagne

One feature of the layers for children's learning proposed by Gagne proceeds chiefly from Signal Learning to Verbal Association, Concept Learning, and Problem Solving. However for adult

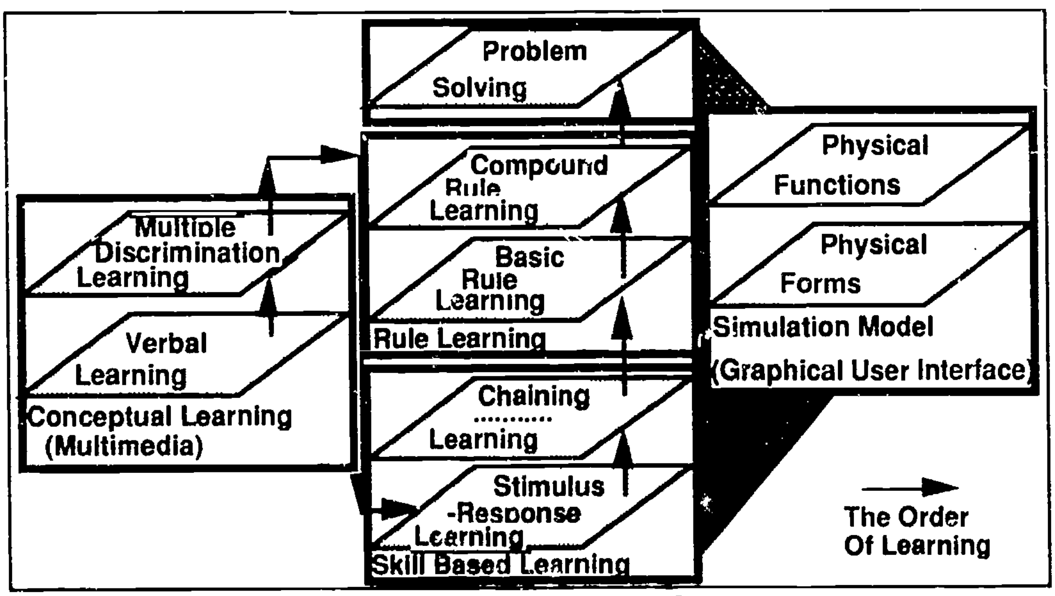


Fig. 2 Hierarchical Layers For Plant Operations

ERIC

training, the concept of the domain should be introduced first of all and verbal learning is usually done at an earlier stage.

2. Proposed Hierarchical Layers

The proposed hierarchical layers for plant operations shown in Fig. 2 are described below. Trainees usually learn from stage (1) to (7) in order.

(1) Verbal Learning

The basic definitions of the technical terms or symbols needed are presented by texts, graphics, and images in direct instructional modes. The concept of the domain is introduced.

(2) Multiple Discrimination Learning

Ways of discriminating among similar or dissimilar situations are presented by texts, graphics, and images in simple model and direct instructional modes. Learning similar situations prevents misconceptions and learning dissimilar situations promotes memorization.

(3) Stimulus-Response Learning

One step turning on/off/adjusting a switch is presented by simulation. This is the first stage in which trainees see simulation models of the real world. At a later stage, this kind of action will be taken without conscious control.

(4) Chaining of Stimulus-Response Learning

One series of steps in turning on/off/adjusting a switch is presented by simulation. The small step principle is provided for Skill-Based Learning. Active reactions and quick checks are required of the trainees.

(5) Rule Learning

The basic operational rules are applied to simple simulation models. Some rules might not appear directly but suggest basic principles. The simulation models and functional rules are increasingly complex from (5) to (7). These stages are provided in model instruction modes. Trainees learn by both inductive and deductive reasoning.

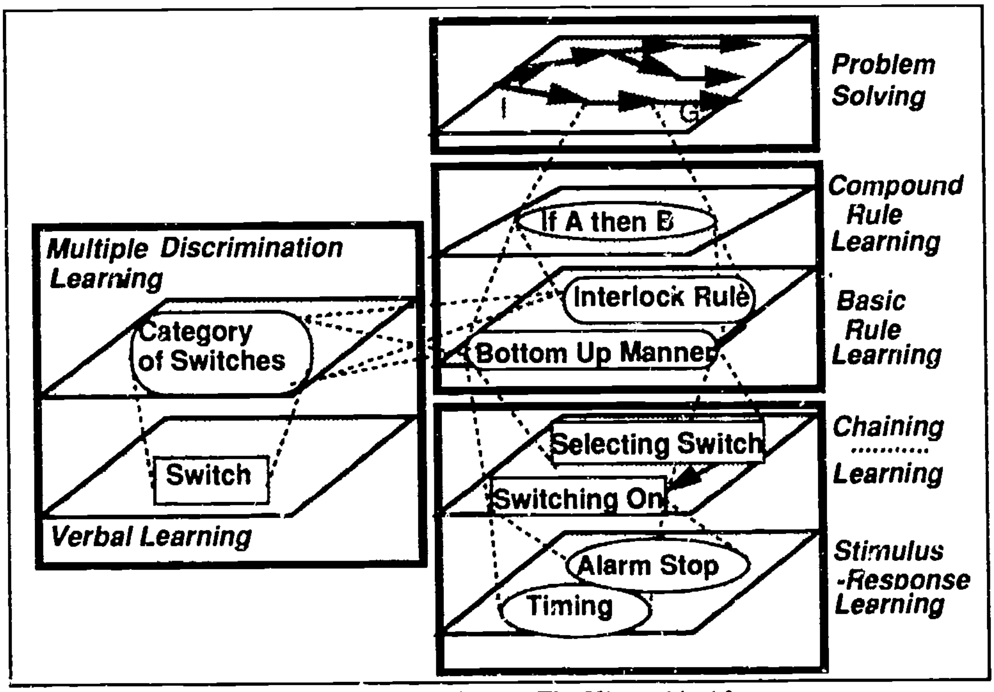


Fig. 3 Relations Among The Hierarchical Layers

(6) Compound Rule Learning

Pragmatic compound operation rules are applied to simple simulation models. They are also used in Problem Solving.

(7) Problem Solving

Trainees operate toward a set goal on pragmatic simulation models. This layer includes the problem solvers. If mixed initiative instruction is included, better training can be provided.

(8) Physical Functions

General functions of the simulation are defined. They were described as class objects. (8) and (9) are not part of the learning process but form the simulation models used from (3) to (7).

(9) Physical Form

This gives the specifications of the simulation model. Trainees contact this layer during operation.

Fig. 3 shows an example of the relationship among specific pieces of knowledge of each stage. This kind of linkage helps both instruction and bug detection. From the perspective of cognitive engineering for human-machine interaction proposed by J. Rasmussen, skill-based behavior, rule-based behavior, and knowledge-based behavior correspond to skill-based learning, rule-based learning and problem solving respectively.

The Features of CETS

In Japan, English education has become more and more important because of attempts at internationalization. Most Japanese people have trouble speaking English although they study English from junior high school. One major reason for this failure is that the school system has focused its attention on reading, translation and grammar. Our system, CETS, is especially for training in speaking.

The main feature of the system is the use of multimedia, including a voice recognition device sold by OGIS Research Institute in Japan. This device has the practical ability to recognize unspecified continuous voice. The purpose of the system is to train students for hearing and speaking in contrast to traditional CAI.

The Structure of CETS

1. Pedagogical stages

The system consists of 3 stages. These are basic practice, pronunciation practice, and conversation

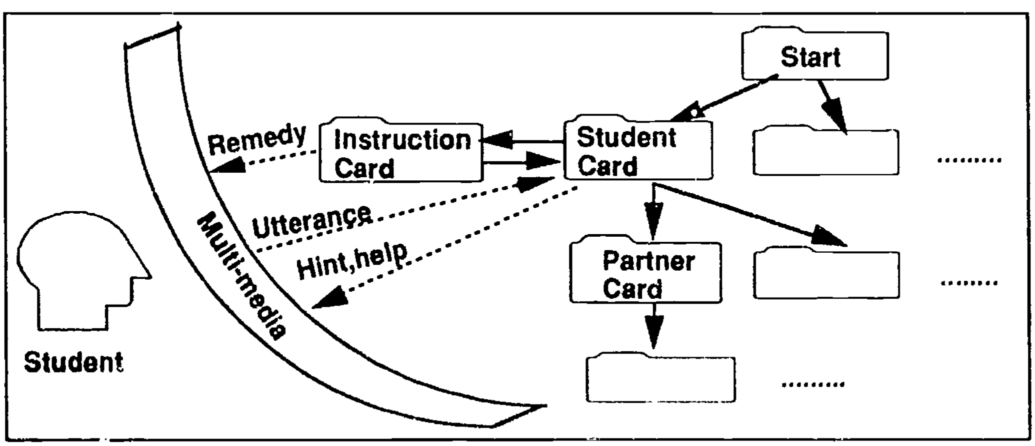


Fig. 4 Hypertext Type Knowledge Base

practice.
(1) Basic Practice
Students learn grammar, vocabulary, and expressions through text. This is now under development.
(2) Pronunciation Practice
Students learn intonation and stress as well as pronunciation through voice recognition. Their results are displayed by graphics in comparison to that of native speakers. Appropriate advice is also provided.
(3) Conversation Practice
Students learn pragmatic conversational English by simulation. The learning proceeds from rote learning to limited free conversation. The conversations are represented with a tree structure. Each node means an utterance or alternative expression, and each link varies the situation. Tree structures are implemented by hypertexts shown in Fig. 4.

2. Internal system

Fig. 5 shows the modules of Conversation Practice.
(1) Hypertext Type Knowledge Base
Conversational knowledge is described in each hypertext linked to the multimedia. The knowledge on student cards including wrong utterances is linked with instruction cards which provide appropriate advice. The knowledge on partner cards, including utterances by persons appearing on the screen, manages the Laser Disk (interactive video). The knowledge base is made using the authoring system by human instructors.
(2) Control Module
The control module provides simulation and the pedagogical remedies managing the knowledge base. The module also selects the teaching materials.

3. User Interface & System Environment

(1) Mouse
Students can select from several menus by mouse.

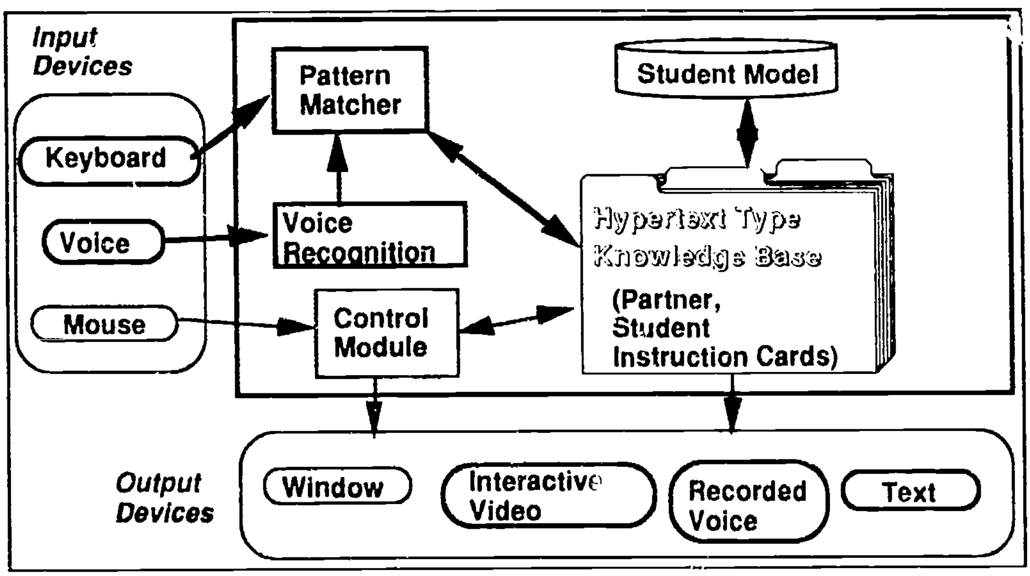


Fig. 5 Conceptual Framework for Conversational Practice

(2) Voice/Keyboard
During training voice is basically the major means of input. If a student's utterance is beyond a certain threshold, the utterance is matched to one of the candidate sentences, if possible by trial and error. Otherwise s/he should speak again or input it by keyboard.
(3) Other Interfaces
Graphics and static/dynamic picture images recorded on a Laser Disk are used for visuals. Statements are represented with text supported by Japanese voice synthesis.
(4) Platform
The system is run on a Sun Workstation with a high resolution color bit-mapped display, a laser disk drive, and a voice recognition device.

The Common Goal of the Systems

CETS can also consist of hierarchical layers similar to those of plant operations. The proposed layers for linguistic learning shown in Fig. 6 are described below. Trainees usually learn from stage (1) to (7) in order.

(1) Verbal Learning
Necessary grammar and vocabulary for conversation are learned
(2) Multiple Discrimination Learning
Ways of discriminating among similar or dissimilar grammar/vocabulary/expressions are learned.
(3) Stimulus-Response Learning
Pronunciation practice of key words for the conversation is provided.
(4) Chaining of Stimulus-Response Learning
Pronunciation practice of key sentences for the conversation is provided.
(5) Rote Learning (introduction of knowledge)
Each model scene is provided by video on a window. Students are expected to be able to speak correctly using simple typical expressions in the scene. (Students memorize the basic expressions and speak smoothly with correct pronunciations.)
(6) Basic Problem Solving (application of knowledge)

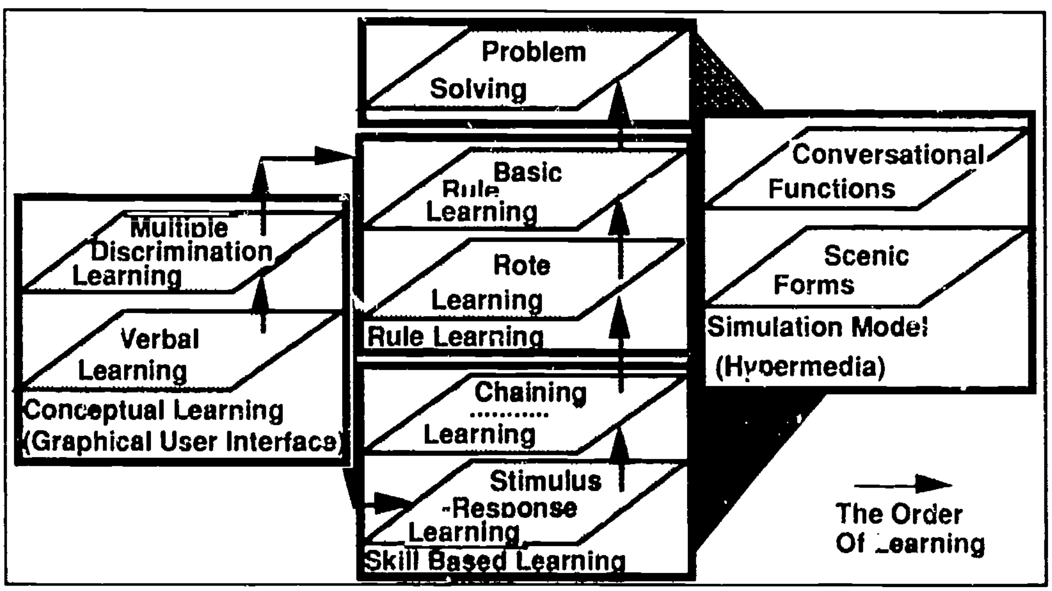


Fig. 6 Hierarchical Layers For Conversational English

A student should be able to speak correctly using simple typical expressions in several scenes. In this case, the system chooses which is the best scene to display and the context is changed according to the student's answer.
(7) Problem Solving (stability of knowledge)
A student should be able to speak correctly in any scene. In this case, alternative expressions are accepted.
(8) Conversational Functions
A sentence, the alternative expression, and the boundary conditions are described. They were shown in Fig. 4 as the student and partner cards. (8) and (9) are not part of the learning process but form the simulation models used in (3) to (7).
(9) Scenic Forms
These are specifications of the simulation model. Trainees contact this layer during conversation.

Conclusion

PCTS has become one of the most well-known examples of ITSs in Japan, and we are contributing significantly to the academic community. It is used at each Osaka Gas plant for the convenience of trainees who can not easily go to the training center because of their work. The authoring system for PCTS will be delivered soon.

CETS is very attractive because of its reality and practical use. CETS has a course whose topic is business appointments. Some coursewares are now being installed.

We will continue to develop the new concept of ITSs based on simulation models and multimedia. We would like to express our gratitude to the project members of Osaka Gas and Oki Electric Co. Ltd. for their cooperation.

References

[1] Sleeman D., B Brown et al., Intelligent Tutoring Systems (Academic Press, USA, 1982).
[2] Hollan J. et al., STEAMER, in AI Magazine (USA, 1987) 15-27.
[3] Towne Douglas M., Allen Munro at al., Representing System Behaviors and Expert Behaviors for Intelligent Tutoring (USC, USA, 1987)
[4] Toyota J. and Y. Nakamura, An overview of knowledge representation and tutoring strategy in intelligent CAI, in: Information Processing (IPSJ, Japan, 1988) Vol.29 No. 11 1266-1274
[5] Kawamura K., J. Bourne, N. Miyasaka, M. Inui et al., Development of a model-based intelligent training system, in: Proc. AI Japan (Japan, 1987) 295-306.
[6] Inui M., N. Miyasaka, K. Kawamura and J. Bourne, Development of a Model-Based Intelligent Training System, in: Future Generation Computer Systems 5 (Elsevier Science Publishing B.V., North-Holland, 1989)
[7] Inui M., N. Miyasaka, Development of a model-based intelligent training system and authoring system, The IEICE, E73, No.3 (Japan, 1990)
[8] Barrett T., A taxonomy of reading comprehension, in: T. Barrett, Teaching in the Middle Grades (Addison Wesley, USA, 1974).
[9] De Kleer J., Causal and Teleological Reasoning in Circuit Recognition (MIT Press, USA, 1977).
[10] Inui M., N. Miyasaka et al., Development of a model-based intelligent training system, in: Advanced Research on Computers in Education (Elsevier Science Publishers B.V., Holland, 1991)
[11] Rasmussen J., Information processing and human-machine interaction (North-Holland, 1986)
[12] Stevens A. L. and Allan Collins, Multiple Conceptual Models
[13] Inui M., Fundamental Research on Simulation Based Intelligent Tutoring Systems, in: The World Congress on Expert Systems Proceeding (USA,1991)

ERIC

The Future of Hypermedia Based Learning Environments: Problems, Prospects, and Entailments

David H. Jonassen
University of Colorado

The assumption that hypermedia has implications for learning remains generally untested. Proponents have claimed that hypermedia information systems would revolutionize the educational process. Little evidence supports this prediction. At least two reasons for the this lack of efficacy exist. First, the level of engagement and purpose among learners in most hypermedia systems does not require meaningful processing of the content. Second, effectively using hypermedia entails the use of learning and metalearning strategies that learners are not skilled in. The assumptions about the ways that learners would use hypermedia and process the information in it have not been empirically supported. This presentation will argue that the most effective hypermedia applications are based on constructivist learning models. Such applications are computer-based, hypermedia learning environments that represent the natural complexity of content domains, support collaborative knowledge construction, and support intentional learning. Rather than merely presenting linked information, hypermedia learning environments foster deeper level thinking about the content being studied. Hypermedia learning environments represent a natural progression of learning technologies from their behaviorist roots, through their cognitive growth into their constructivist facilitation of meaningful learning.

I shall begin this presentation with a chronology of computer-based technologies which describes the progression of technologies, their conceptual bases, and applications of them (Fig. 1). Next, I shall describe the range of information processing engaged by these environments (Fig. 2). I shall demonstrate examples of such environments that have been designed and researched at the University of Colorado. Finally I will reflect on the problems and prospects entailed by these environments. For example, from the perspectives of social expectations and adoption/diffusion, the efficacy of some computer based innovations may be questionable. Ironically, the design of many of these environments espouse situating the learning experience in the context of some real world task. Had the design and development of these environments been situated in the real world context of the classroom, as designers, we may recognize that our innovations have far exceeded our ability to implement them in such a context. These and other arguments will hopefully generate discussion about the problems and prospects of these environments in terms of the activities, responsibilities, abilities and expectation of the learning environments, teachers, and learners.

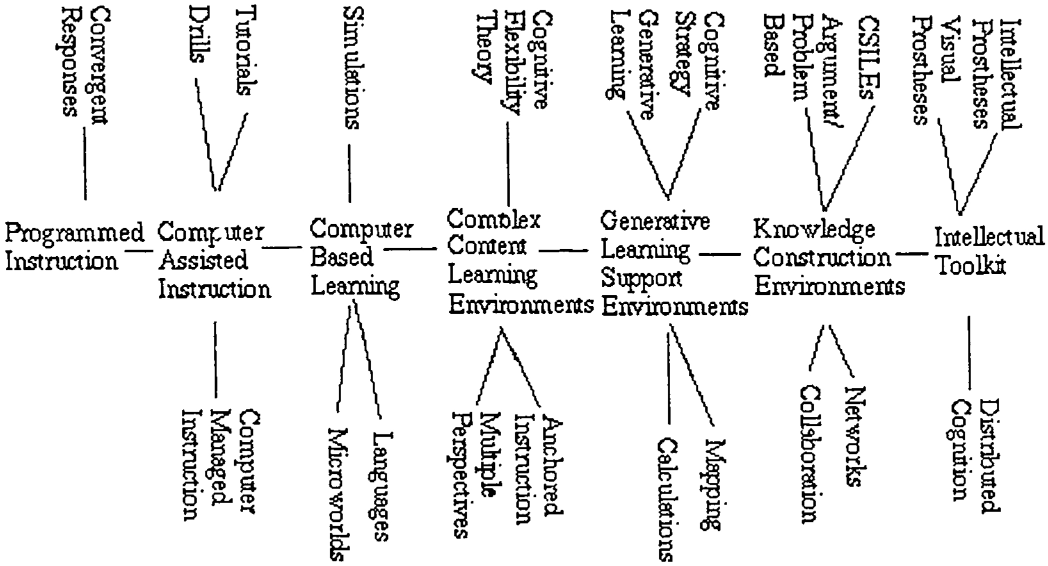


Fig 1. Chronology of computer-based technologies.

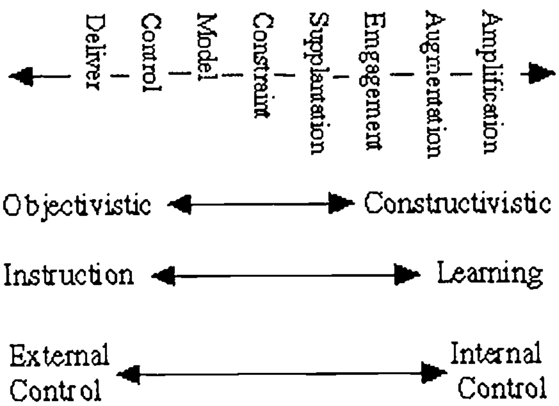


Fig. 2. Information processing engaged by technologies.

Hypermedia System Functionality: Strategies for Entangling Users

CHARLES J. KACMAR
Department of Computer Science
Florida State University - Tallahassee, FL 32306 USA

Introduction

Collaborative and groupware hypermedia systems provide environments which support goal-oriented human-human interaction tasks in electronic environments (Ellis, Gibbs, & Rein, 1991). These systems are designed by studying human interaction and modeling their interactions through system features or by applying principles of human interaction to computerized settings. The architecture of these systems is usually tailored to specific problem domains by supporting only the relevant conceptual, data, and procedural models needed for group activity.

gIBIS (Conklin & Begeman, 1988), VNS (Shipman, Chaney, & Gorry, 1989), and WE (Smith, Weiss, & Ferguson, 1987) are examples of domain-specific collaborative hypermedia systems. Characteristics include shared links and interfaces tailored to a particular task environment.

KMS (Akscyn, McCracken, & Yoder, 1988) and Intermedia (Yankelovich, Haan, Meyrowitz, & Drucker, 1988) are examples of collaborative hypermedia systems which are not specifically designed to support a particular group interaction model. These *general purpose* systems derive their power from the flexibility they offer users in performing a variety of collaborative activities within a single computing environment. Collaboration is supported as a result of their architectures and interactions with services provided by the underlying computing environments.

The remaining sections of this paper present the results of three case studies involving the use of a general purpose collaborative hypermedia system to support an argumentation/discussion task. The results of these studies provide additional insight into hypertext usability and user behavior by studying the contents of hypertexts relative to the interface and functional characteristics of systems.

Collaboration Project Overview

Background

Subjects. The subjects in the case studies were primarily graduate students enrolled in a graduate course on computer-human interaction and hypertext/hypermedia. The first case study was conducted at Rensselaer Polytechnic Institute. The remaining case studies were conducted at Florida State University. Case study 1 involved 13 students, case study 2 involved 12 students, and case study 3 involved 5 students. The students in case study 3 also participated in case study 2. Two students in study 1 were not computer science majors, 1 student in study 2 was not a computer science major - all students in study 3 were computer science majors.

Preparation. Prior to the task, students attended course lectures and performed various course-related assignments. Typical assignments included reading journal articles and books, writing papers, presenting material in class, watching video tapes of hypermedia systems, and using and evaluating hypermedia and non-hypermedia systems. Class lectures and assignments provided students at least 6-8 weeks of exposure to hypermedia materials and at least 2 different experiences with hypermedia systems prior to being assigned the collaboration task.

Task. The task required students to discuss and/or argue about topics. Students selected the topic in the first and third case studies. The topic of discussion was assigned to the students in the second study. No boundaries were given on the task and in fact the students were encouraged to comment and link any information in the hypertext. Students were given approximately 4 weeks to complete the task.

The first group of students identified 8 topics of interest. After approximately 2 weeks of "discussion," the most popular topic concerned environmental issues. The second group of students was assigned the topic of "halloween." This topic was selected due to interest in local protests and news coverage of controversial festivities associated with this event. The third group selected the topic of "spring break."

Measures

The primary goals of the task were to provide students with experience in using a general purpose hypermedia system and to obtain comments about the features of the system for the purposes of enhancing its capabilities. Consequently, a rigorous or controlled experimental environment was not employed during the first study. However, it became apparent about midway through the second group of students that a pattern of behavior, similar to that exhibited in the first group, had developed. The pattern related to the structure and content of the hypertexts used and created by the group. Even though the purpose of the task was initially non-experimental, it was decided to observe and note these patterns, especially since these students were reasonably competent in the field. Hence, the importance of this research concerns the outcome of the hypertexts and the visual characteristics of the nodes that were created.

Method

Subjects invoked the system and followed a sequence of links to enter the hypertext for the task. The base node contained a list of topics to be discussed which were usually defined with text. The directed graph associated with each topic was navigated in a forward direction by clicking the left mouse button on a link marker. Clicking the right mouse button allowed the user to backtrack through the hypertext. Although the graph structure was primarily hierarchical, the system allowed arbitrary link connections.

A node also was reachable using a "GoTo" facility. A menu item initiated the goto request and allowed the user to identify the name of a node to which to navigate. After the user entered the name of the node, the system searched for the object and navigated to the node if it existed. If the node did not exist, the application created a new node having the name specified and navigated to the node. Node names were displayed in the title bar of the display window.

Environment

System overview. We now describe the operation and interface of the system. The purpose is to provide a framework for understanding the tools available to the students in conjunction with performing the task.

The system used in the case studies is a prototype, single-process implementation of the PROXHY system (Kacmar & Leggett, 1991). It is a general purpose collaborative hypermedia system, supporting various types of objects and is not tailored toward a particular human or group interaction model. The system runs on Sun workstations and allows users at various locations to simultaneously access and modify a shared hypertext. Hypertexts are stored on a file server and managed using standard system file services.

Interface. A "painter's canvas" window with direct manipulation (Shneiderman, 1992) and a spatial metaphor supports the drawing of all objects. The user display consists of three components: the main display window, menu, and message window. Various sizes and fonts of text are supported. Other objects supported by the system include: line, circle/ellipse, square/rectangle, polygon, bitmap, and patterns. Text objects associated with links are bounded by rectangles. A dashed border indicates link markers for graphical objects. Linked bitmaps appear as 3D-buttons.

The spatial interface allows text objects to be created by moving the mouse cursor to any location in the main window and typing. Graphical objects are created by first clicking on a menu item to identify the type of the object and then indicating the location and size/shape of the object with subsequent mouse clicks. Step-by-step instructions for creating graphical objects appear in the message window.

Links are directly attached to objects, they do not have to be attached to "buttons." Bitmaps can be used as the basis for a button but most users choose to use text as the primary endpoint of the source end of a link. Nodes are the most preferred destination of a link. Graphical objects are primarily used to emphasize aspects of the discussion or for drawing.

Two methods of creating links are provided. A "quick link" is created by twice clicking the left mouse button on an object. This causes the object to be selected, a new node created, and then a link to be created which connects the object to the new node. A second method allows links to connect one or more existing objects in the same or different nodes. For each endpoint of a link, the objects that constitute the link end are identified by selection and an anchor is attached to the group. A link connects the anchors. Anchor and link deletion is provided through menu selection and identification (pointing) of the link marker(s) to be affected by the operation.

Data model and management. All objects have a unique identity. Primitive objects are named by the system. Nodes can be named by users or are named by the system. Anchors and links are implemented as first-class objects. Objects are stored in directories with anchors and links residing in the same directory as the application objects. The data management facilities provide complete transparency from the directory structures of the hypertexts. Most users never name any of the objects they

ERIC

create or access.

The hypermedia data model is the traditional anchor-link-anchor model. Anchors are attached to one or more objects at the source and destination ends of a link to provide 1-to-1, 1-to-N, M-to-1, or M-to-N relationships. Anchors may be attached to nodes or primitive objects within nodes thus providing node-to-node, node-to-primitive, primitive-to-node, or primitive-to-primitive associations. Link navigation is uni-directional.

Support for collaboration. The system uses network file services to support collaboration. Objects created by one user are accessible by another when a user creates objects under the appropriate file access privileges. Users at different workstations may access the same hypertext at the same time. The typical mode of operation for group interaction is unrestricted group access rights for all users.

Observations and Interpretations

The interesting aspects of this project concern the behavior of the students while using the system and the state of the hypertexts during and after the tasks. Although browsing the hypertexts did not result in any notable difficulties, all three groups encountered problems during authoring. Problems related to using the system were resolved through class and individual discussions. Obviously, some individuals experienced more difficulty than others.

Results

Chunking. One of the major difficulties observed among the study groups concerned the structuring and placing of information in the hypertexts. On several occasions, students requested a set of "rules" or "guidelines" for deciding the amount of information to place in a node and where to attach links. Given the fact that students had spent 6-8 weeks prior to the task studying and using this and other hypermedia systems, these skill-directed experiences did not seem to provide sufficient models of structuring and linking for the argumentation task.

Link traversal. Even though the quick link facility provided a fast and simple mechanism for creating links, some students indicated that navigating a link was "too great a distance" depending on the topic and content of the item to which they wanted to comment.

Proximity. Another interesting phenomenon concerns the quantity and location of objects in the nodes. Nodes were heavily populated with a variety of information and object types. The depth of the hypertexts was "shallow" - between 2 and 4 levels (links). Since the spatial interface allowed the students to "place" their arguments or comments wherever they desired, comments were generally placed *as close as possible* to the comment which initiated their response. In some cases, the response was positioned adjacent to or overlaying the original comment. The debriefing indicated that *proximity* was very important and that the spatial interface metaphor supported this behavior.

Presentation and link markers. Links were usually attached to text objects at the source end of a link and connected to nodes at the destination. As the quantity of objects in a node increased, the use of graphical objects as source-end objects increased. In some nodes, graphical objects became the dominant object to define the link source.

The application allowed objects to overlay each other. This resulted in a situation where linked text objects were overlayed with graphical objects associated with a different link. A linked text object was visible to the user even if it was overlayed with a transparent (non-filled), linked, graphical object. Users clicking on the area containing the intersection of the text and graphical objects, expecting to navigate to the destination defined by the text object, became confused when the link associated with the graphical object was followed. Not surprisingly, this resulted in significant disorientation for most users.

To circumvent this situation, users temporarily moved the graphical object out of the way in order to follow the link associated with the text object. After backtracking to the point of departure, the graphical object was repositioned to its original location, overlaying the text object. Intersecting, multi-linked objects were eventually reorganized by the students such that the graphical object either was moved away from the text object or the two objects were linked to a common node from which two points of departure were defined.

Students quickly learned to use graphical objects to "entice" others to follow their links. Objects drawn in the shape of arrows were commonly used to guide a user to a related discussion (as in a guided tour (see Trigg (1988) and Marshall & Irish (1989) for a related discussion). Border thickness and fill patterns also were used to increase enticement.

Link types. Three link behaviors were supported: reference, include, and automatic. The existence of multiple linking mechanisms created many problems for all three groups. A common error resulted in nodes being overlayed with other nodes. This situation is caused by defining an automatic include link which includes the entire contents of another node.

When a "deformed" node results, disorientation is an immediate consequence. According to some students, the first few times they encountered nodes of this type their reaction was to click the "Quit" menu item to exit from the system because they thought the system had failed. Others simply navigated away from the deformed node and then returned to the same location using a different route to bypass the included link. When a deformed node could not be avoided, "warning signs" were posted to warn others about these nodes during navigation.

Discussion

The observations from these studies indicate that authoring a hypertext and using a collaborative hypertext system can be difficult for average experienced users. This is significant since hypertext facilities are often touted by vendors and researchers as simplifying human-computer interaction and are especially suitable for novice users. Although these claims might be true for browsing-oriented systems, they are not true for systems which provide more functionality. Difficulties can occur when node granularity and the boundaries between nodes are left to the individual author. Disorientation may result (Conklin, 1987; Nielsen, 1990; Utting & Yankelovich, 1989), especially when the underlying hypertext is continually changing as in a collaborative environment.

Browsing and studying existing hypermedia systems seems to provide little help in forming the necessary conceptual models for understanding and organizing information. To create a usable hypertext, authors must be trained as to the amount of information to place in a node and how information is to be structured (De Young, 1990). As discussed, some users had noticeable difficulty authoring even after many weeks of classroom and hands-on experience.

Multiple anchor and link types are powerful and effective facilities for creating hypertexts when used appropriately. However, these and other mechanisms can be troublesome since there is no "natural" organization for a particular hypertext. If it is important to use these facilities in hypertexts, authors must be trained in using these facilities and must fully understand their effects on the hypertext and navigation. Iterative design, authoring, and evaluation methods for both the systems and their databases are needed to ensure usable systems (Eveland & Bikson, 1988).

The prototype did not provide assistance in managing group activities. Users collided when attempting to update the same nodes or objects simultaneously. Coordination methods, similar to those found in database systems, are needed to protect the integrity of the hypertexts (Ellis, Gibbs, & Rein, 1991; Malcolm, Poltrock, & Schuler, 1991). Other coordination methods also should be established prior to group activity to minimize group conflicts. One of the most requested enhancements was a display which graphically depicted the current users and their locations in the hypertexts.

Summary

Authoring hypertexts can be difficult, even for *average*-experienced users. The most basic difficulties concern structuring the hypertext and quantifying the amount of information to place in a node. Although the amount of information in a node may be limited by the size of the display and manner in which the node is presented, users at this experience level appear to have conceptual difficulties which transcend display characteristics.

One of the most important findings concerns object proximity. Although a hypermedia system may provide a very simple and fast mechanism for creating links, nodes may become overpopulated as users attempt to position information as close as possible. This can result in nodes which contain different ideas and/or are illogically chunked; and, hypertexts which are flat or have no obvious logical structure.

References

Akscyn, R., McCracken, D. & Yoder, E. (1988). KMS: A distributed hypermedia system for managing knowledge in organizations. *Communications of the ACM*, 31 (7), 820-835.

Conklin, J. (1987). Hypertext: An introduction and survey. *IEEE Computer*, 20 (9), 17-41.

Conklin, J. & Begeman, M. (1988). *gIBIS: A Tool for All Reasons*. MCC Technical Report #STP-252-88. Microelectronics and Computer Technology Corporation, Austin, TX.

De Young, L. (1990). Linking considered harmful. *Proceedings of the European*

Conference on Hypertext, (INRIA, France, November). In *Hypertext: Concepts, Systems, and Applications*, Eds. A. Rizk, N. Streitz, and J. André, Cambridge University Press, Cambridge, England, pp. 238-249.

Ellis, C., Gibbs, S. & Rein, G. (1991). Groupware: Some issues and experiences. *Communications of the ACM*, 34 (1), 39-58.

Eveland, J. & Bikson, T. (1988). Work group structures and computer support: A field experiment. *Proceedings of the Conference on Computer-Supported Cooperative Work*, (Portland, OR, September 26-28), pp. 324-343.

Kacmar, C. & Leggett, J. (1991). PROXHY: A process-oriented extensible hypertext architecture. *ACM Transactions on Information Systems*, 9 (4), 399-419.

Malcolm, K., Poltrock, S. & Schuler, D. (1991). Industrial strength hypermedia: Requirements for a large engineering enterprise. *Hypertext '91 Proceedings*, (San Antonio, TX, December 15-18), pp. 13-24.

Marshall, C. & Irish, P. (1989). Guided tours and on-line presentations: How authors make existing hypertext intelligible for readers. *Hypertext '89 Proceedings*, (Pittsburgh, PA, November 5-8), pp. 15-26.

Nielsen, J. (1990). The art of navigating through hypertext. *Communications of the ACM*, 33, 3 (March), 297-310.

Shipman, F., Chaney, R. & Gorry, G. (1989). Distributed hypertext for collaborative research: The Virtual Notebook system. *Hypertext '89 Proceedings*, (Pittsburgh, PA, November 5-8), pp. 129-135.

Shneiderman, B. (1992). *Designing the User Interface*. Second Edition. Addison-Wesley, Reading, MA.

Smith, J., Weiss, S. & Ferguson, G. (1987). A hypertext writing environment and its cognitive basis. *Hypertext '87 Proceedings*, (Chapel Hill, NC, November 13-15), pp. 195-214.

Trigg, R. (1988). Guided tours and tabletops: Tools for communicating in a hypertext environment. *Proceedings of the Conference on Computer-Supported Cooperative Work*, (Portland, OR, September 26-28), pp. 216-226.

Utting, K. & Yankelovich, N. (1989). Context and orientation in hypermedia networks. *ACM Transactions on Information Systems*, 7 (1), 58-84.

Yankelovich, N., Haan, B., Meyrowitz, N. & Drucker, S. (1988). Intermedia: The concept and construction of a seamless information environment. *IEEE Computer*, 21 (1), 81-96.

ERIC

Diagrammatic Literacy: A Rudimentary Multimedia Approach

Raymond P. Kirsch
LaSalle University
Mathematical Sciences
Philadelphia, PA 19141 USA
kirsch@aca.lasalle.edu

Introduction

Diagrams are an aid to communication. They provide pictorial descriptions of information and concepts from an application domain. When a diagram is shown during the communication act, the information is more likely to be remembered and understood than if that information were conveyed without the diagram (Levin, 1987).

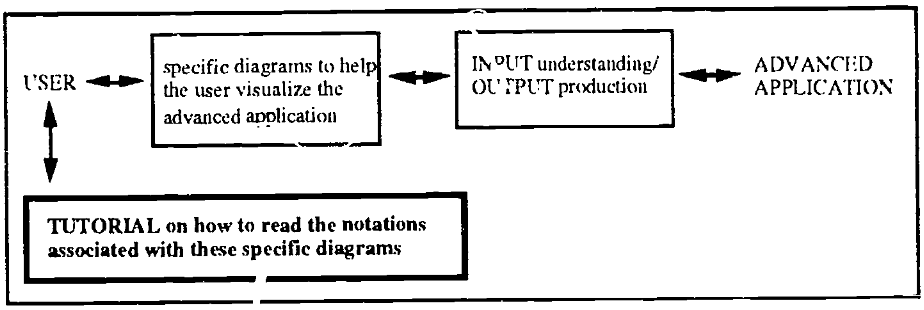


Figure 1. Typical User Interface Components with Suggested Improvement.

Many graphical user interfaces make use of diagrams, and have some form of the components appearing across the top of Figure 1. For example, dataflow diagrams help people think about the software system design task, or the picture of a database schema diagram is shown to communicate exactly how a database is structured. The user is freed from having to remember details, and is likely to be more productive and accurate. In some instances, diagrams are the primary language with which concepts are communicated between the user and the computer (DiBattista, 1990). For example, in QBD* (Angelaccio, 1990), the schema diagram itself is used to retrieve information from a database.

Specific diagrams are aimed at helping the user to visualize advanced applications. But, are these diagrams understood by the average user? Very little, if any, instruction on the reading and understanding of diagrams is provided during a person's formal education -- diagrammatic literacy is an assumed skill. As things are currently, unsophisticated users must consult an off-line source for this information. But many users will not take the initiative to research fundamental diagrammatic notations and will make assumptions. This leads to potential misunderstanding and misconceptions, and certainly not to the communication and understanding, which the diagrams are expected to facilitate.

Figure 1 also shows an improvement. A user working with the advanced application who finds the need for a review of its specific diagrammatic notations can request a rudimentary multimedia tutorial. Afterwards, that user is more likely to understand the advanced application.

There is a need for on-line presentations of the basic notations associated with domain specific diagrams. The primary goal of the research reported in this paper is to provide a tool to help produce them.

Problem

The central focus this research is the production of rudimentary multimedia presentations aimed at diagrammatic literacy. Primary objectives are the design of these presentations, and the design of a system that helps to produce them.

Examination of the processes involved in understanding diagrams has led to three questions

1) What skills are required to be able to "read diagrams," and how are these skills taught?

2) How should computer-based presentations that explain a diagram be constructed?

3) Can a system be designed which reduces the complexity associated with authoring these presentations?

Research into the first question has provided the background for addressing questions two and three. This paper highlights the main features of a solution to the problems posed by questions two and three. For a detailed description see (Kirsch, 1992).

Background

Diagrams are composite images which are comprised of components and composite images. Bertin (1983) argues that comprehending diagrams occurs in a part to whole manner. Comprehension of the components is prerequisite to understanding a composite image which those components make up. For example, knowing what the schema diagram components

each represent is prerequisite to understanding the composite image

Bertin also suggests that there are three successive levels in the development of graphical comprehension, and that there are progressive stages within each level. Paivio (1986), Edwards (1979), and Saint-Martin (1989) suggest that visual literacy is a right-brain, and fundamentally nonverbal process. Elaboration Theory of Instruction (Reigeluth, 1980) correlates with Bertin's curricular suggestions, and adds that each level of presentation should be introduced and reviewed. A confluence of these influences 1) provides a theoretical, rather than ad hoc, framework for presentation design, and 2) results in the presentation design requirements appearing in Figure 2. These requirements have directed the design of the system described in the remainder of this paper.

Three Presentation Levels

Elementary -- simple composite images
Intermediate -- composite images with groups of components
Advanced -- whole composite images

For each level, proceed according to the following five steps

1) Introduce the level of presentation

2) Present prerequisite components which appear in the composite images of step 3

For each component

a) Identify the component in general

i) show a general picture
ii) play descriptive audio(s)

b) show example(s)

3) Present composite images appropriate to this level

For each composite image

a) Identify the composite image in general

i) show a general picture
ii) play descriptive audio(s)

b) show example(s)

4) Review the composite images in terms of the current level

For each composite image presented in step 3, show an animated-walk through which highlights concepts from the current level

5) Review in terms of prior levels

For each composite image presented in step 3, show an animated-walk through that highlights concepts on the current and prior levels

Figure 2. Presentation Design Requirements.

The System

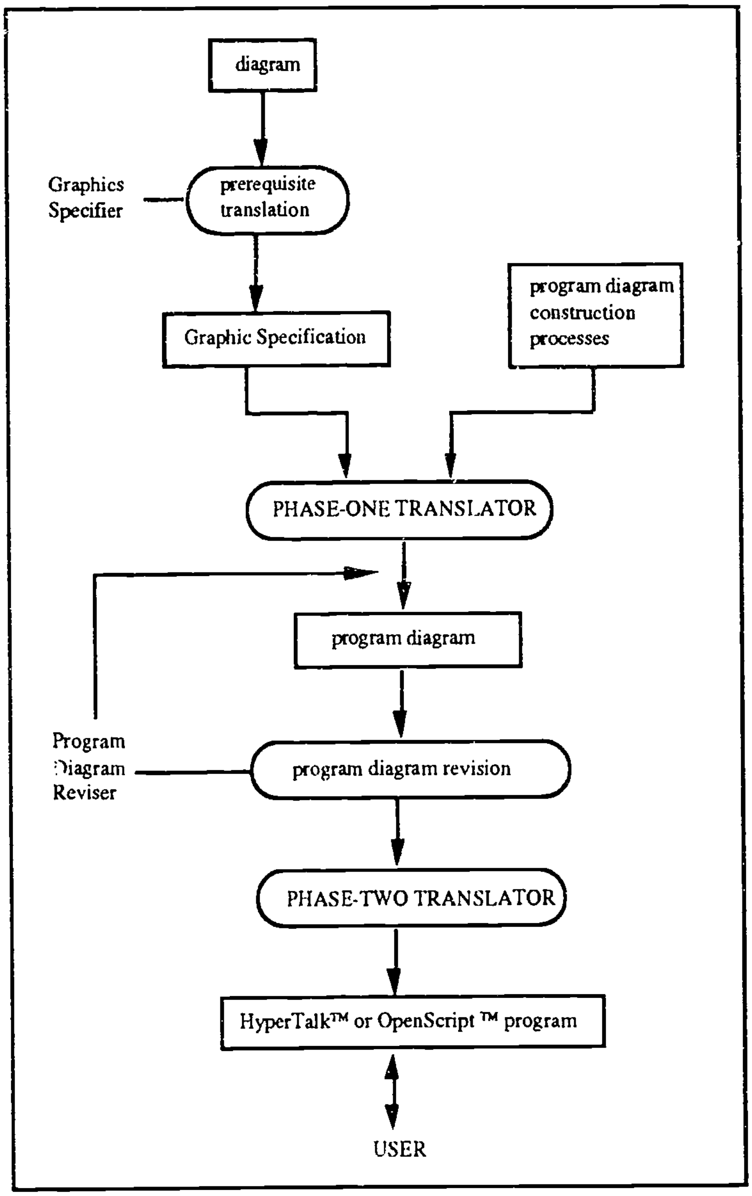


Figure 3. The Complete System.

The presentation design requirements in Figure 2 were used to design the system shown in Figure 3. This system translates a diagram into a rudimentary multimedia presentation on the graphical notations associated with that diagram. Translation begins with prerequisite translation, a process which is accomplished by a person, who is referred to as the Graphics Specifier. Example diagrams are examined to determine components, composite images, and to provide descriptive annotations that identify them. The components, composite images, and corresponding annotations are then sequenced according to the presentation design requirements (Figure 2). The result is written in a form known as the Graphic Specification.

The Graphic Specification is input to the Phase-One Translator which automatically produces a default program diagram. Program diagram construction processes direct this translation phase. The default program diagram is an internal representation of the presentation which can either be executed as is (following Phase-Two translation), or can be revised by a person who is referred to as the Program Diagram Reviser.

Phase-Two translation converts a program diagram into an executable script in a target multimedia language, e g., in HyperTalk™ or OpenScript™. When the script is executed, a rudimentary multimedia tutorial is presented to the user. After witnessing this presentation, the user may select portions for review.

Walk-Through

As a very simple illustration, the systematic translation of a diagram into a presentation might begin with the Graphics Specifier examining at the following example diagram

to come up with a) PICTURE values, which are vector graphic objects that generalize the components and composite images appearing in the example, and b) ANNOTATIONS, which are the general textual descriptions typically used when speaking about these items. Resulting values are entered into Component and Composite Image Tables at the appropriate level, which in this case is elementary, or level one.

LEVEL	PICTURE	ANNOTATION
1	<label>	entity class
1	<label>	scalar class
1	<label>	single-valued attribute

Figure 4. Sample Component Table

LEVEL	PICTURE	ANNOTATION
1	<label> <label> <label>	single-valued scalar attribute
1	<label> <label> <label>	single-valued entity attribute

Figure 5. Sample Composite Image Table

In actual practice, a wide range of examples would be examined by the Graphics Specifier to generalize all of the typical components, composite images, and descriptive annotations for a

2 9 5

ERIC

specific diagram type. These would be entered into the Graphic Specification according to increasing level of sophistication (Figure 2).

Continuing with the example, the Phase-One Translator uses the values and sequencing provided by the portion of the Graphic Specification in Figure 4 to produce the program diagram snippet in Figure 6. Another portion of the program diagram (not shown) is produced using the values and sequencing in Figure 5. The details of program diagram form are beyond the scope of this paper (see Kirsch, 1992).

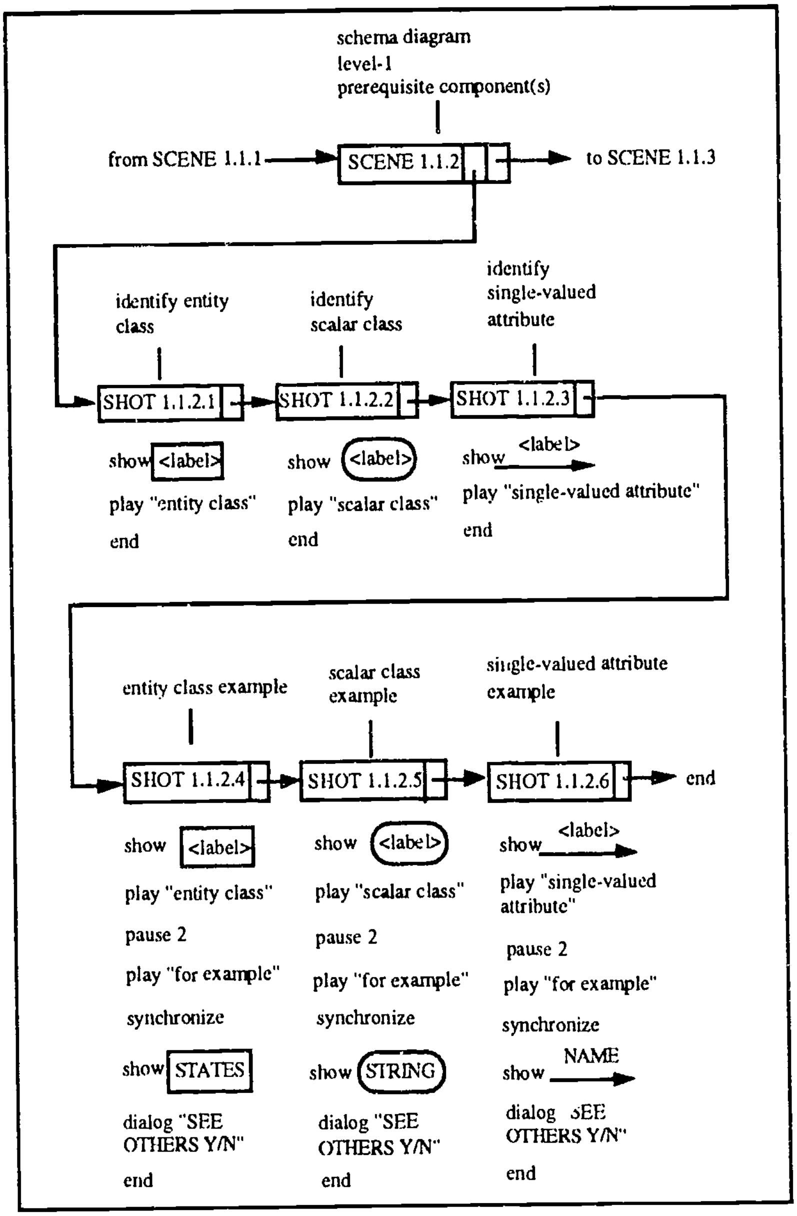


Figure 6. Sample Output from Phase-One Translation.

ERIC

The program diagram is a starting point for presentation authoring, or it may be passed unaltered to the Phase-Two translator. The Phase-Two translator traverses the program diagram, and as each SHOT node is visited, the statements found there are translated into one or more executable statements in the target multimedia language, e.g., HyperTalk™ or OpenScript™. The resulting statements are appended to a script, which when executed, provides the rudimentary multimedia tutorial. To conclude the example, the three components in Figure 4 are identified (defined in Figure 2, step 2) when the scripted version of the program diagram in Figure 6 is executed.

Summary

The need for on-line tutorial assistance with understanding diagrams has provided the motivation for theoretically-based presentation design requirements, and has resulted in the design of a system to assist in the production of rudimentary multimedia presentations aimed at diagrammatic literacy. The system design, as outlined in this paper, is complete, and is capable of producing tutorials for database schema diagrams, and for dataflow diagrams. While the system has not been implemented, experiments using both HyperCard™ and ToolBook™, and walk-throughs of all essential parts, indicate that implementation is feasible. Developing a prototype is the next step on this project.

References

Angelaccio, M., Catarci, T., & Santucci, G. (1990). QBD*: a fully visual system for E-R oriented databases. Proceedings 1989 IEEE Workshop on Visual Languages.

Bertin, J. (1983). Semiology of graphics: Diagrams, networks, maps. (W. Berg, Trans.). London: University of Wisconsin Press. (Original work published 1967).

DiBattista, G., Giammarco, A., Santucci, G, & Tamassia, R. (1990). The architecture of diagram server. Proceedings 1990 IEEE Workshop on Visual Languages.

Edwards, B. (1979). Drawing on the right side of the brain. Los Angeles: J.P. Tarcher Inc.

Kirsch, R. P. (1992). Computer-based teaching of visual literacy. Unpublished doctoral dissertation, Temple University, Philadelphia, PA.

Levin, J. R., Anglin, G.J. & Carney, R.N. (1987). On empirically validating functions of pictures in prose. In D. M. Willows & H. A. Houghton (Eds.), The psychology of illustration: Volume 1. New York: Springer-Verlag.

Piavio, A. (1986). Mental representations: A dual coding approach. New York: Oxford University Press.

Reigeluth, C.M., Merrill M.D., Wilson, B.G., & Spiller, R.T. (1980). The elaboration theory of instruction: A model for sequencing and synthesizing instruction. Instructional Science, 9(3), 195-219.

Saint-Martin, F. (1989). From visible to visual language: Artificial intelligence and visual semiology. Semiotica , 77(1-3), 303-16.

Improving Visual Programming Languages for Multimedia Authoring

JOHN F. KOEGEL AND JESSE M. HEINES
Department of Computer Science
University of Massachusetts–Lowell
Lowell, MA 01854, USA
{koegel,heines}@cs.ulowell.edu

1. Overview

1.1 Multimedia Authoring

The creation and editing of interactive computer-based presentations which combine text, graphics, audio, and video is called multimedia authoring. Contemporary tools provide both media editing and multimedia composition as well as more specialized services including database and file access, courseware support, and extensibility. Such tools make it possible to design sophisticated presentations, but are more difficult to use than previous menu and text systems for several reasons:

- The more powerful authoring environments require greater learning time
- The creation and integration of animations, video, and audio is a more elaborate process and is typically less familiar to authors than text and graphics composition

We have been actively developing and using multimedia authoring tools for a number of years (Koegel, Rutledge & Heines, 1992). This paper summarizes our observations and experience with two commercial multimedia authoring packages and provides the following results:

- Detailed criteria for evaluating visual programming interfaces for multimedia authoring
- Example iconic programs for comparing different multimedia visual programming tools
- General recommendations for improving the design of such systems' user interfaces

Multimedia authoring tools are a significant enabling tool for courseware and other interactive presentations. However, the development of such materials with the best tools available today requires significant expertise, equipment, and time. The goal of this paper is to further advance the usability of such tools.

1.2 Synopsis of Authoring Paradigms

There are at least fifty commercial packages for developing computer-based training or interactive presentations. Many of these provide audio-visual device control for multimedia delivery. The most common paradigms for multimedia authoring are:

- **Outline:** An outline of the presentation is constructed in a text-based outlining editor. Each outline entry can be expanded into a presentation screen which incorporates graphics, text, and interaction.
- **Visual Programming:** A set of icons are arranged in a graph which specifies the interactions and control path for the presentation. The functionality associated with each icon can be modified using associated menus and editors. Typically a simple text language is available for performing calculations within an icon.
- **Scriptware:** Graphics, text, and other media editors are combined with an integrated and special purpose programming language. Programs (scripts) define the control flow and interaction behavior. The script language is typically intended to have a simple, easy to use syntax.

Shu (Shu, 1988) defines visual programming as "the use of meaningful graphic representations in the process of programming". To us, however, visual programming is closer to Shu's definition of a visual programming language: "a language which uses some visual representations to accomplish what would otherwise have to be written in a traditional one-dimensional programming language". This definition is still quite broad, and indeed must be narrowed for clarification in the current context. In this paper, we use visual programming to mean the construction of a graph of interconnected icons which can be interpreted by an execution engine to perform a series of tasks.

Iconic authoring systems are often mistakenly considered to be object-oriented because some systems call icons objects, and virtually all object-oriented systems use icons. To clarify matters: an icon is a graphical representation of a function, while an object is a programming entity made up of a type definition *plus* definitions of the operations that can be performed on objects of that type. In addition, object-oriented systems allow the definition of general properties of a class of objects plus specific properties of subclasses of that object. For example, in an object-oriented authoring system one might define the general properties of a user input action and then distinguish the specific properties of mouse, keyboard, and audio input. Thus, while some iconic authoring systems are *built* using object-oriented technology, that is, they are implemented using an object-oriented language, their appearance to multimedia authors is iconic, not object-oriented. Iconic authoring systems use icons to represent fully contained functions that can be combined in various ways to create multimedia programs.

2. Review and Comparison

2.1 Package Selection and General Criteria

AimTech IconAuthor (AimTech, 1991) and *Authorware Professional* (Authorware, 1989) are both widely used and highly-rated commercial multimedia authoring tools. Each is available on multiple platforms and is suitable for developing applications for training or presentations. Both use the visual programming paradigm, but with notable differences. For these reasons, we selected these two packages for study.

We are primarily interested in evaluating and improving the user interface of multimedia authoring tools which employ visual programming techniques. We do not consider many issues that would be important in selecting one of these packages for actual use, such as integrated tools, performance, platform, or data interchange. Instead, we focus on two areas: the semantics of the icon set and the human factors of the iconic interface.

Icon set semantics concern identifying the meaning of the primitives and constructors defined by the visual programming language, including individual functionality and compositional methods. Programming language theory has several systems for developing formal semantics of a programming language. These techniques are used to develop systematic answers to questions related to functionality and correctness. Visual languages which closely follow procedural languages, or which can be translated into a procedural language, will borrow their semantics from the corresponding procedural model. *The practical consequence of semantic analysis is to answer questions such as what presentation and interaction sequences can be represented by the language, and whether all iconic compositions have a consistent interpretation.* Additionally, multimedia authoring tools include methods for temporal composition and synchronization; the correctness, expressiveness, and precision of these facilities is perhaps a unique aspect of multimedia authoring languages when compared to traditional programming languages.

In visual programming, the user manipulates a pictorial representation of a sequence of actions to achieve some larger function. Many factors influence the power and usability of such an interface, including the consistency and complexity of the icons, the graph organization, and the number of steps needed to perform editing operations. The domain of multimedia authoring, because of its visual orientation, adds the relation between the iconic graph and the visual aspects of the presentation. We are particularly interested in the relationship between the graph construction and editing activity and the multimedia authoring process.

Table 1: Functionality and Representation Power

Measure	IconAuthor	Authorware
Icon Set Size	Large	Small
Icon Set Functions	Mixture of primitives and composition icons	Multi-function control icons
Icon Set Semantics	HLL-like; flow-chart	Special
Icon Set Extensibility	Extensible	Not extensible
Hierarchy and abstraction	Discretionary; Multiple techniques: composite, subroutine, module	Forced: non-scrollable windows; inability to expand interaction icons vertically; map icon
Family Style	Flow Chart	Special
Semantics of Composition	Follows High-Level Language	1. Loop in Interaction Icon 2. Composition of Sequencer and Interaction
Relationship of presentation structure to graph structure	Low structural relation	Low structural relation

2.2 IconAuthor

IconAuthor provides a visual programming editor in which the control icons closely correspond to typical programming language functions such as loop, if-then, and subroutine. The icons are single function, with several composition icons provided to build reusable collections of icons. Each icon's function is accessed through an associated content editor. The content editor for every icon has the same appearance, making it easy to use. There is a drag-and-drop window which contains the current graph; several graphs can be opened simultaneously. Editing functions are available through a set of pull-down menus; an icon ribbon provides quick access to frequently used functions. A set of integrated tools is included for text, graphics, animation, image editing, and video control. These tools are invocable from either the associated icon or the pull-down menus.

2.2.1 Function and Semantics

The icons are single function, with several composition icons provided to build reusable collections of icons. These are grouped in the following families:

- **Flow:** branches, if-then, loop, menu, module
- **Input:** keyboard input, mouse input
- **Output:** audio, structured graphics, erasure, graphics attributes
- **Data:** database and file access, system variables, built-in math and string functions

- **Multimedia:** audio, video card, video player
- **Extensions:** DDE and DDL interfaces, subroutine, a help facility, RS-232 control
- **Custom:** user-defined icons

Table 2: Human Factors of Visual Language

Item	IconAuthor	Authorware
Visual Consistency	High	Very high
Visual complexity and inspectability	Because icons are more primitive in function, complex graphs can result. This can be alleviated by use of modules and composites	Low for individual windows, but can be high for deeply nested maps
Graph size and relation to presentation function	If composites are used, graph size can be kept low	Average; many standard functions can be programmed using maps nested to only 1 or 2 levels
Use of color	Color used for icons and background; some customization available	Color not used in graph construction
Text Labeling	Label icons	Label graph
Number of interaction steps needed to create something	Slightly high, because indirect access to smart text and graphics editors through icon content editors	Average, many choices provided in easy-to-use dialog boxes, especially for the interaction icon
Number of interaction steps needed to modify something	Slightly high, because of indirect access to certain editors	Average, with some types of interactions difficult to conceptualize, but we see this as a standard visual program problem, not a unique characteristic of Authorware
Debugging, in particular how debugging operations such as trace and breakpoint are integrated with the interface	Difficult to find currently executing icon; can selectively enable and disable icons and groups of icons	Good ability to jump to the icon controlling the current display, but no single step capability; excellent facility to insert stop and stop "flags" to control partial execution, but only one pair of flags can be inserted for any one run; can't skip over or disable icons
Support for authoring process	Various icons for hierarchical design, and some predefined composite icons	Good, uses "models" as program templates for standard operations, also has ability to insert empty map icons as "placeholders" that will be completed later

2.2.2 Human Factors

There are a large number of icons, but because the icons are primitive in function, they are easy to learn. The tool does not enforce any specific programming discipline, so it is quite possible for an author to create complex unstructured graphs. Because the icons are primitive in function, graphs tend to be large and hard to inspect. Although the tool provides techniques for managing the size of the graph, the author must still think at a relatively low-level of icon function during editing.

The graph area can be scaled, making it easier to navigate large graphs. Groups of icons can also be collapsed or expanded as desired, helping to control graph complexity. The icon ribbon makes a number of frequently used operations quite accessible. IconAuthor provides an extensive on-line help facility.

2.3 Authorware

Authorware is a visual programming system for Windows and Macintosh systems. It is based on PCD3, a project originally undertaken at Control Data Corporation. *Authorware* uses a flowchart visual programming paradigm. The author drags icons from an icon palette onto a flow line to specify the flow of control

ERIC

through the program. Each icon has a number of parameters that are set through dialog boxes, to which the author gains access by double clicking the icon. A distinctive feature is the low number (seven) and apparent simplicity of its icons, although this simplicity can be deceptive due to the large number of options provided through dialog boxes. *Authorware* supports standard multimedia extensions, smoothly integrating them into the visual programming system.

While *Authorware* provides a wide array of facilities, it is difficult to extend. The attractiveness of this system is in its high degree of attention to human factors and the consequent elegance of its implementation. It is easy to learn and a pleasure to use. Its main drawback is one shared by all visual programming systems: when programs become complex, the difficulty in managing the icon network increases geometrically.

2.3.1 Function and Semantics

Authorware is based on a procedural model. Its seven icons are: display, erase, wait, calc, interaction, decision, and map. Of these, the **interaction** and **decision** icons are the most misleading, as they provide a huge amount of functionality through extensive use of options specified in dialog boxes. In addition, these two icons provide icon sequences that tie together combinations of the other five icons in a structured manner. For example, the **interaction** icon can be used to accept mouse or keyboard input, analyze click locations, strings, and numbers, and provide conditional feedback for any number of alternatives. The **decision** icon can be used to control program flow in a simple if/then manner or through timed or numbered iterations.

The **calc** icon is also deceptive. It can indeed be used to set and retrieve values of variables and perform computations, but it also acts as the interface to other programs via DDE and DDL. We have seen some *Authorware* programs that are up to 30% **calc** icons, which somewhat defeats the purpose of the visual programming paradigm. The syntax of statements in **calc** icons is C-like, but unfortunately only 424 characters are allowed, making it difficult to write substantial routines.

Authorware contains a number of sophisticated options on icon functionality that can be difficult for the new user to grasp. For example, calculations can be "attached" to other icons without the need for **calc** icons. The functionality of an **interaction** icon includes a required display, which can be annoying when one wants to enter began an interaction sequence without changing the current display. The order in which interaction tests are performed is critical, and can cause interesting problems in some complex interaction sequences.

All in all, these semantic difficulties are not difficult to master; they just take time. To a programmer used to straightforward textual function calls, however, the specification of visual algorithms can be tedious. We believe this to be a characteristic of all production visual languages, and one that needs to be addressed through additional study.

2.3.2 Human Factors

Authorware is particularly strong in the area of human factors. With only seven icons, the system has a distinctly "approachable" feel, especially to novice users. It is an easy system to try things out in, as undos are trivial. The system has an exceptional degree of consistency and is very smoothly integrated with its underlying window system (either Windows or the Macintosh). The system designers have taken particular care to address novice computer users with such characteristics as allowing spaces in variables names. The main problems we see in *Authorware* are:

- It forces a strict hierarchical structure (only one icon is allowed below interaction choices, although this icon may be a map icon)

- Windows cannot be scrolled vertically and are therefore limited to a maximum of about nine icons along the main flow line (although any of these may be map icons which may be nested to any depth)
- **calc** fields only allow 424 characters and the grammar of **if** statements allows only one statement in the **then** and **else** clauses
- **maps** are not true subroutines (each time a map is inserted, a copy of the code is inserted; therefore, changing a visual algorithm that is used repeatedly may require that change to be made in many places)
- No on-line help facility

2.4 Summary

Tables 1 and 2 summarizes the comparison of these two tools for the categories of Functionality and Human Factors.

3. Benchmarks

In this section we present one example benchmark for evaluating different multimedia authoring tools. Benchmarks that we have used to compare tools include:

- Sequential presentation
- Menu-based presentation
- Animated model
- Interactive hypertext and hypermedia

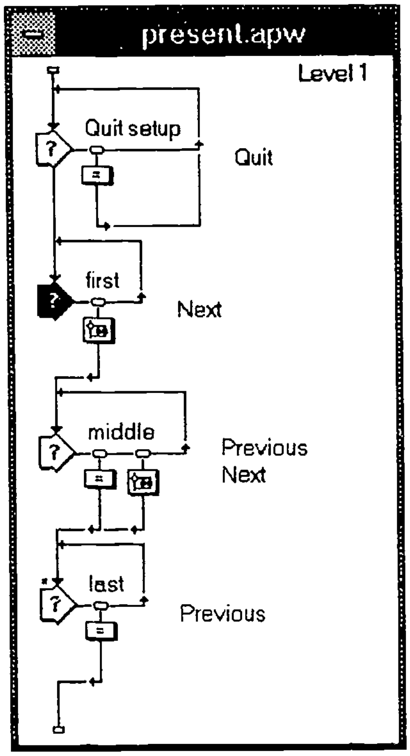
3.1 Graph Structure for a Sequential Presentation: Authorware Professional

The example is a simple program to present slides and provide the ability to move forward and backward through the slides or quite at any time. The *Authorware* structure of such a program for 3 slides is shown in Figure 1. The slide content is placed in the **Interaction** icons (the arrow-like icons labelled with question marks in the figure). The first interaction, labelled "Quit setup," sets up a permanent Quit button, which appears on all slides and which, when pressed, exits the slide presentation.

The **Interaction** icon labelled "first" (and highlighted in the figure) contains one push button labelled Next. When this button is pushed, control passes to the **map** icon directly under the "first" label, after which it exits the first interaction and enters the next one. In this case, the **map** icon is empty. (You need to have *something* for the interaction icon to do, and an empty **map** icon fills this syntactic need.)

The icon labelled "middle" has two push buttons, labelled Previous and Next, respectively. The one labelled Next has an empty **map** icon and works just like that in the previous interaction. The one labelled Previous, however, has to explicitly jump to the previous icon, so a **calc** icon (the one labelled with an =

FIGURE 1. Sequential Presentation in Authorware Professional



303

sign) is used containing the statement:

GoTo(IconID@"first")

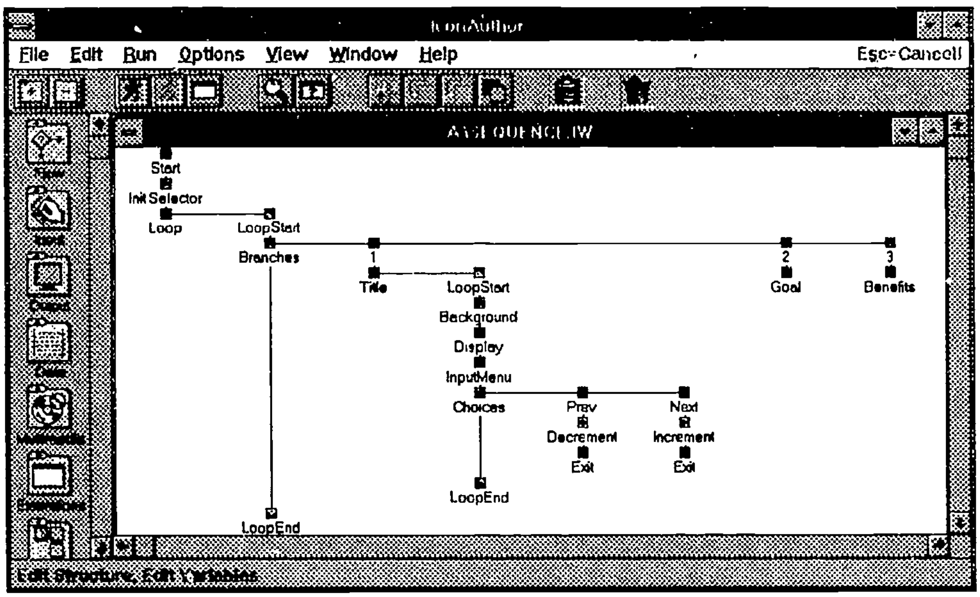
This syntax is a bit cumbersome—and certainly not visual—especially since the name of the icon to branch to must be a string *constant* and cannot be a variable. However, Authorware is does update icon names embedded in calc icons if you happen to change the name of the icon. When we cut and paste icons to enlarge the slide show, however, we must open the **calc** icon and explicitly change the name of the previous interaction icon to go to. We don't have to do this for the Next button, because its **map** icon is simply a place holder and the built-in interaction icon functionality automatically goes to the next icon when the interaction exits.

3.2 Graph Structure for Sequential Presentation: IconAuthor

The sequential presentation in *IconAuthor* (Figure 2) relies upon an index variable to specify the screen to display. In the graph shown in Figure 2, successive screens are numbered 1, 2, 3, ...; at any point in the presentation, a branch forward is performed by incrementing the index variable and a branch backwards is performed by decrementing the index variable. Each screen has an equivalent sub-graph structure, as shown for the first screen, where the graph has been expanded. After the screen contents are presented, two choices are activated. Whichever choice is selected leads to the index variable being updated and control returns to the outer-most loop. For presentations with many slides, the branch lists should be broken into segments to facilitate insertion and deletion.

The mechanism for branching is awkward because it relies upon numeric labeling. Insertion and deletion of screens requires re-editing of indices. Some type of symbolic branching support would be a significant improvement.

FIGURE 2. Sequential presentation in AimTech IconAuthor



4. Summary

The visual programming paradigm presents a number of fascinating human factors problems, but none of these seems strong enough to stem the tide of interest in this appealing technique. We believe that the best visual programming systems will be those that allow a high degree of user customization to address these problems, such as defining one's own composite building blocks with their own icons. In addition, these systems must allow users to break out to standard computer languages when textual approaches are more efficient. Visual programming has great promise for multimedia authoring, but we do not want to be locked into using this paradigm when others are more efficient.

5. References

AimTech Corporation. (1991). IconAuthor User Manual. Nashua, NH.

Authorware, Inc. (1989). Authorware Professional Manual. Redwood City, CA.

Chang, S.-K. (1987). Visual Languages: A Tutorial and a Survey. *IEEE Software*. Jan. 1987. pp. 29-39.

Koegel, J., Rutledge, J. and Heines, J. (1992). Visual Programming Abstractions for Interactive Multimedia Presentation Authoring. *Proceedings of the 1992 IEEE Workshop on Visual Languages.*

Shu, Nan C. (1988). *Visual Programming*. Van Nostrand Reinhold: NY.

The Advent of (Standardized) Integrated Multimedia Environments

JOHN F. KOEGEL
Department of Computer Science
University of Massachusetts–Lowell
Lowell, MA 01854, USA
koegel@cs.ulowell.edu

Next generation computer and communications technology for instructional environments is expected to accelerate progress in distributed and collaborative learning and interactive multimedia courseware. Various industry and international standards groups are currently creating specifications which will define a framework for how this technology will be structured. These standards encompass distributed object facilities, hypermedia document architectures, multimedia interchange, scripting languages, media formats, application toolkits, and multimedia networking. In this paper we review these developments, focusing on the overall framework and architecture that will result and its potential impact on delivery of instruction and educational services.

1. The Media-Enabled Classroom and the Media-Oriented Student

The technology for low-cost multimedia systems is intersecting with the next generation products and services of the computer, telecommunications, and consumer entertainment industries. The direct impact on education includes richer instructional materials (Apple, 1991), more powerful presentation tools (Koegel & Heines, 1993), and access to geographically distributed educational services and databases. The indirect effects will also be considerable, including a student population which is not only media-conversant but is media-oriented due to daily use of interactive television, videophones, sophisticated video games, and hand-held personal organizers.

The challenge for educators involves both mastering these new technologies and tools and creating a classroom where students are not primarily tool-users and presentation viewers but are creators, gatherers and organizers of multimedia materials. The challenge for technologists will be to provide a foundation for educators and others on which to build new content and application services. Various multimedia standards currently being developed will be a significant part of this foundation.

While multimedia technology has been appropriated by the classroom for some time, current practice is typically confined to the stand-alone multimedia computer. With digital technology, distributed multimedia computing will become the status quo, bringing with it:

- The ability to easily store and access on-line audio, pictures, and video from any point on the network
- Computer-mediated collaboration using shared applications and desktop video conferencing
- Computer-based telephony, i.e., the integration of computer communication tools with the on-site PBX and the public switched network
- Powerful mobile computing devices, which connect to the distributed computing environment through mobile LAN and/or cellular telecommunications and which work as personal organizers, locators, and entertainment controls

Even for environments which are accustomed to LAN-based distributed computing, the large scale of communications will be a new dimension. Possible educational uses include:

- Collaboration between students working at home

ERIC

- Classroom-based field trips to museums, industries, and sites around the world
- Interaction with special education facilities at other schools or universities

Computer applications will be more accessible to all age groups since multimedia interfaces reduce the barrier to entry for information access by replacing menus and text with visuals and speech. Further, multimedia technology can be expected to enhance a number of other trends in educational computing, including:

- Intelligent tutoring systems (Parkes & Self, 1990)
- Inspectible environments (diSessa, 1986)
- Distance learning (Friedman & Jurkat, 1992)
- Electronic books (Backer, 1985)
- Large-scale hypermedia systems (van Dam, 1987)

In the remainder of the paper we summarize the current directions in distributed multimedia computing and communications, focusing on specifying an overall framework from which the various services can be viewed.

2. Frameworks for Viewing the Evolution of Multimedia Technology

2.1 Multimedia Communications

During the past decade, the telecommunications industry has developed a roadmap for the evolution of digital switched communication services based on ISDN (Roca, 1985). ISDN standardizes both connection interfaces and transmission protocols. Consequently ISDN interfaces can be provided by both computer systems and private branch exchanges (PBXs). The goal of ISDN is to provide distributed digital communication services in a location-independent manner. To achieve this, the signaling protocol of ISDN permits PBXs, the public switched network, and other ISDN-compatible equipment to work as a unit. As a result, the local PBX and computer network can be switching and service nodes in the ISDN network.

In addition to ISDN, several other technology trends are enabling the convergence of desktop computing and traditional communications services. First, in many companies personal computers and workstations are found on most desks because of the need for even non-computer users to communicate via electronic mail. Second, digital audio hardware is now affordable and is beginning to be a part of the standard configuration of desktop computers. Third, there are now a variety of products that permit companies to configure PBXs using third-party boards and standard computers so that the PBX can be more easily integrated with the computer network.

There are a number of practical reasons for the merger of computer and telephony communications. The current practice of separate systems for voice and data communications causes disjoint information. For example, voice mail can not be easily archived and accessed with related electronic mail. The practice of accessing communication services via touch-tone keypads is much more tedious and time consuming than access via computer-based user interfaces. Further, related time management, meeting management, and groupware software applications are computer-based and could be much more useful when integrated with voice communications.

A recent CCITT standard for video telephony on ISDN networks (Liou, 1990) known as H.261 or p*64 is one of the first steps towards digital multimedia communications. H.261 defines a scalable compressed digital video encoding at two different resolutions which can operate over the entire range of ISDN (multiples of 64 kbps up to about 1.5 Mbps). The video compression algorithm is similar to that used by ISO MPEG (Le Gall, 1991), but the audio encoding for H.261 (CCITT G.711) is different than that defined by MPEG. Nevertheless it appears likely that H.261 will be used for video conferencing in both ISDN and computer networks.

The need for higher quality video transmission for full-motion video applications and the availability of high-bandwidth fiber optic telecommunications are leading to a successor of ISDN known as broadband

ISDN (B-ISDN). B-ISDN is still in development, but will likely use a high-speed switching technology known as asynchronous transfer mode (ATM) that provides two orders of magnitude faster switching speed than ISDN (Natarajan & Slawsky, 1992). ATM is the basis for the forthcoming metropolitan area network (MAN) standard and future ATM computer interfaces are being planned by a number of computer workstation vendors.

The emergence of mobile communication technologies is the final point of intersection with telecommunications services. Portable computers are now available with cellular telephone interfaces, permitting remote access to computer networks. Additionally, a number of new handheld computing devices called personal digital assistants (PDAs) will have network interfaces so that users can access services through workstations on LANs. Currently the bandwidths for such connections are limited and the devices are restricted in terms of computational performance because of size considerations. These end-points will most likely be the final stage in the distribution of multimedia communications.

The growing connectivity (Figure 1) between computer and public switched networks will make greater integration of function possible. For the computer user, workstation telephony will provide real-time video conferencing, call forwarding, and directory services. These functions will be integrated with software applications such as personal and group calendars, meeting schedulers, and personnel locators. Additionally video conferencing and application sharing between groups of users will be combined to provide multimedia conferencing. As mobile computing and communication matures, multimedia conferencing will be possible on a global scale.

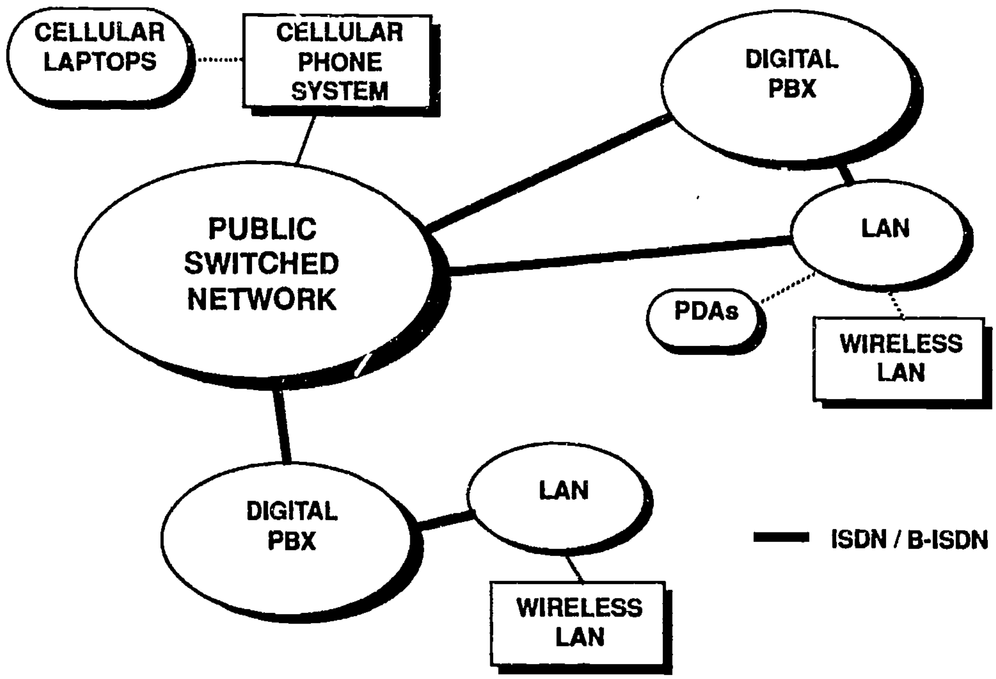


FIGURE 1. Intersection of telephony with computer communications will involve both ISDN and B-ISDN standards as well as new technologies for mobile computing

2.2 Multimedia Computing

Multimedia computing is migrating from a standalone model to a distributed model. Additionally, vendors are working together in industry trade groups such as the Interactive Multimedia Association (IMA, 1992) and international standards bodies to define a common framework and services for multimedia applications. Many in the industry realize that standard formats and application programmer interfaces will accelerate the acceptance of multimedia technology. Figure 2 shows a layered framework for these services.

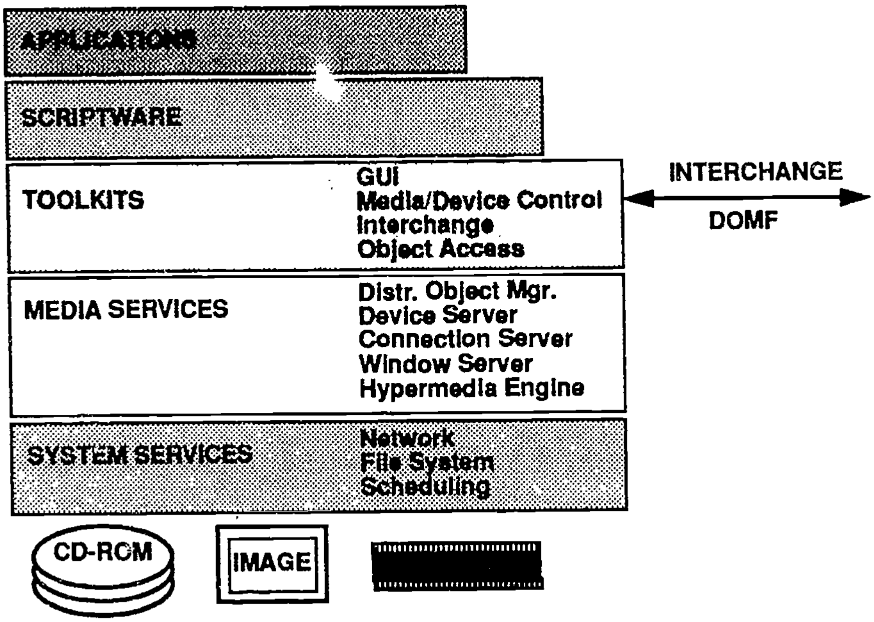


FIGURE 2. Multimedia Services Framework

A scripting language is a special purpose programming language used by application developers and end users to create multimedia materials. Such languages provide more sophisticated control capabilities than other authoring techniques, and are designed to be easier to use than conventional programming languages. Currently each multimedia authoring product defines its own scripting language, making it difficult for multimedia materials to be interchanged. Several proposals to define a standard scripting language are underway (Table 1) (IMA, 1992). These languages are likely to be object-oriented and allow access to various services normally provided to application developers through toolkits.

The services provided by applications toolkits will be significantly extended to support multimedia applications development (Koegel, Keskin & Rutledge, 1992). New functionality will include temporal composition and synchronization, image processing (Butler & Krolak, 1991), device control (IMA, 1992), and multimedia interchange (Koegel, 1992). A number of proposals for multimedia interchange formats are being developed (Table 1). These formats permit multimedia presentations to be exchanged between different platforms and applications. Additionally, toolkits will provide applications with access to the services of other applications (Figure 3) using a distributed object management facility (DOMF).

DOMF is object-oriented mechanism by which applications export functionality and data for use by other applications. The OMG standard for DOMF (Object Management Group, 1991) permits all participating applications to access available services in a network-transparent manner. The user will be able to build dynamic documents which reference data in any of a variety of applications, from multimedia authoring tools to spreadsheets to databases. This is an extension to the hypertext concept known as hyperapplication.

Below the toolkit layer are a set of media and system services that the toolkits access. The device server provides an abstract device control model for multimedia peripherals such as codecs, laser disc players, special effects equipment, and synthesizers. The connection server manages connections for desktop video conferencing. The hypermedia engine interprets hyperlink references and resolves object addresses in hypermedia documents. Several standards efforts address these services (Table 1). In particular, the HyTime standard defines a structuring language for hypermedia documents (ISO 1992, Newcomb 1991) based upon

BEST COPY AVAILABLE
ERIC

the ISO SGML text document model. A HyTime engine provides generic hypermedia services to a variety of multimedia applications (Table 2) (Koegel, Rutledge & Keskin 1993).

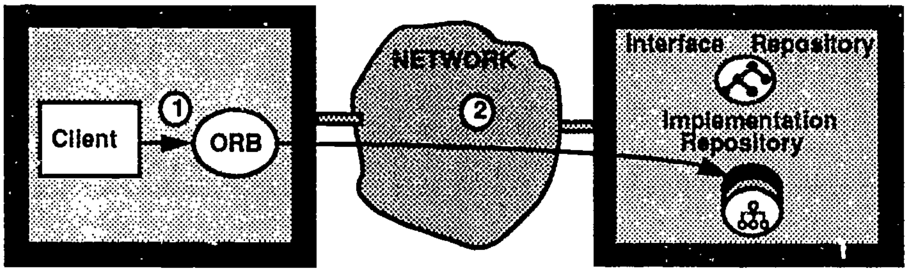


FIGURE 3. The OMG model for a distributed object-oriented application environment provides network transparent object locating and invocation. The client requests a service by contacting the local object request broker (ORB). The ORB locates the service on behalf of the client and then invokes the appropriate method using the implementation repository.

Finally, a number of multimedia formats have been standardized to provide the basic building blocks of multimedia information storage and delivery. These include the ISO digital video compression standard MPEG (Le Gall, 1991) and the ISO still image compression standard JPEG (Wallace, 1991).

TABLE 1. Multimedia standards can be viewed as layers in an overall framework (rows) or by source (columns)

USER	FUNCTION	ISO / CCITT	TRADE GROUP	UENDOR
END USER	Group Conferencing			
	Multimedia Authoring			
AUTHOR	Scripting Language	SMSL	IMA RFT	Kaleida ScriptX
	Hypermedia Document Architecture	HyTime SGML		
DEVELOPER	Distributed Object Arch.		OMG CORBA	MS OLE
	UI Toolkits	PIKS	X Consortium XIE	Apple QuickTime MS MME
SYSTEM VENDOR	Multimedia System Services		IMA RFT UNIX Intl	
	Multimedia Mail		IETF MIME	
	Interchange Format	MHEG	IMA RFT OMFI	Apple QuickTime Movie File Format
PUBLISHER	Storage Formats	9660	Rocky Ridge	Kodak PhotoCD Philips CD-I
	Media Formats	MPEG JPEG H.261	MIDI	Intel DVI

BEST COPY AVAILABLE

For the educator, these developments will lead to more powerful authoring tools and a growing set of multimedia resources. Multimedia materials will be easily exchanged and re-used in a variety of applications. Multimedia authoring tools will take advantage of a distributed collection of application services and new toolkit facilities to provide powerful interactive multimedia interfaces.

Table 2: HyTime engine facilities vs. application/presentation facilities (Koegel, Rutledge, Rutledge & Keskin, 1993)

Sample HyTime Engine Functions	Sample Application/Presentation Functions
Hyperlink resolution	User input processing
Object addressing	DTD element interpretation
Parsing of measures and schedules	Presentation scheduling
Schedule and dimension transformation	Media rendering

2.3 Consumer Devices

The home market is potentially the largest near-term market for multimedia, with applications like video-on-demand, home-shopping, and video games. So far the technology has been developed with a notable lack of standards, creating fragmentation. For example, all the home multimedia systems have incompatible operating systems and compact disk formats (Table 2). These systems have the potential to impact educational computing in two ways. First, a growing number of educational titles are being offered for use on these consumer devices, making them valuable reference materials for the library or classroom. Second, any non-standard formats that reach a significant volume will probably be supported by the computer industry.

Table 3: Representative consumer devices that connect to the home television for multimedia presentation

Vendor	Operating System	CD Format
Philips CD-I	OS9	CD-I
Commodore CD TV	AmigaDos	CD TV
Tandy VIS	Modular Windows	CD-ROM
Kodak Photo CD	proprietary	Photo CD

3. Competing Trends

The frameworks presented in the previous section have several points of potential intersection. A perspective on this overlap and its potential resolution can be obtained by looking at the competing trends occurring in the convergence of these industries. These competing trends include:

311

Fiber-based CATV vs. B-ISDN: The cable television industry is moving from an analog video distribution system towards optical fiber-based digital video to provide more channels and support for interactive services. The telephone industry is moving from a switched analog audio telephony network to a switched digital video telephony network. The CATV network is non-switched and tree-structured, the broadcast station being the root. ISDN and B-ISDN are digital switched networks that support N-way connections.

Local vs. Distributed Content: High-density compact disk storage is available at a low-cost and can be accessed from both computers and consumer devices. These disks are suitable for static information distribution and can satisfy the requirements for multimedia information delivery. With the emergence of high-speed networks, it will be possible to access geographically remote libraries of multimedia information. Information providers will have the choice of distributing via compact disks or electronic access.

Local vs. Distributed Services: New mobile computing devices (PDAs) will be used as personal organizers, managing phone lists, schedules, notes, and communications. With the intersection of telephony and desktop computing, personal management application software will be integrated with communication interfaces. Consequently mobile computer users will be able to choose between resident application software and access to remote services.

Home Computers vs. Interactive TV Devices: Interactive entertainment devices are powerful dedicated computers with easy-to-use interfaces. Devices such as Commodore CD TV or Tandy VIS may potentially be used to run general purpose applications software written for associated personal computer product. Next generation multimedia computers will provide overlapping functionality with the interactive consumer devices.

4. Summary

There are significant technical, regulatory, and market issues facing the arrival of integrated multimedia environments, but the frameworks for multimedia computing and multimedia communications are well understood. The process by which these systems are realized will be incremental, and various multimedia computing and communications standards will play an important role in the pace of market growth because of the importance of interchangeability to multimedia information use. There are a number of competing trends which are yet to be resolved. However, the two major functions enabled by multimedia systems are unambiguous: richer information processing and delivery and video-based real-time group collaboration.

Technology which is already in the home and used by students will inevitably end up in the educational process. Consequently educational multimedia computing will involve technology from both the consumer entertainment and computer areas. Educators will face challenges such as mastering the multimedia capture and production techniques, working with incompatible media formats, and obtaining copyrighted source materials. Perhaps more significant is the goal is to enable students to be media purveyors as opposed to media viewers.

5. References

Apple Computer Inc. (1991). Visual Almanac. Multimedia Lab Technical Report. 1991.

Backer, D. (1985). Prototype for the Electronic Book. *Media For a Technological Future--Electronic Publishing Plus* (ed. M. Greenberger). Knowledge Industry Publications, White Plains, NY 1985.

Butler, T. and Krolak, P. (1991). An Overview of the Programmer's Imaging Kernel (PIK) Proposed Standard. *Comput. & Graphics* (15) 4.

diSessa, A. A., and Abelson, H. (1986). Boxer: A Reco structible Computation Media. *Communications of the ACM*. 29(9).
Friedman, E., and Jurkat, M. (1992). Computers in Mathematics: Teacher In-Service Training at a Distance. *Enabling Technologies for High-Bandwidth Applications, SPIE* vol. 1785.
Interactive Multimedia Association. (1992). Requests for Technologies.
ISO. (1992) ISO/IEC IS 10744, Hypermedia/Time-based Document Structuring Language (HyTime), August 1992.
ISO. (1993) ISO/IEC JTC 1/SC 29/WG 12 Multimedia and Hypermedia information encoding Expert Group (MHEG). Working Draft 1. Jan. 1993.
Koegel, J. (1992). On the Design of Multimedia Interchange Formats. *Proc. 3rd Intl. Workshop on Operating System and Network Support for Digital Audio and Video*.
Koegel, J., and Heines, J. (1993). Improving Visual Programming Languages for Multimedia Authoring. *World Conference on Educational Multimedia and Hypermedia.*
Koegel, J., Keskin, C., and Rutledge, J. (1992). Toolkits for Multimedia Interface Design. *Xhibition 92.*
Koegel, J., Rutledge, J., and Heines, J. (1992). Visual Programming Abstractions for Interactive Multimedia Presentation Authoring. *Proceedings of the 1992 IEEE Workshop on Visual Languages.*
Koegel, J. Rutledge, L., Rutledge, J., and Keskin, C. (1993). HyOctane: A HyTime Engine for an MMIS. submitted.
Le Gall, D. (1991). MPEG: A Video Compression Standard for Multimedia Applications. *Communications of the ACM*. (34) 4.
Liou, M. L. (1990). Visual Telephony as an ISDN Application. *IEEE Communications Magazine* (28) 2.
MIT X Consortium (1993). *X Image Extension Protocol Reference Manual. Version 4.12*. O'Reilly & Assoc., Jan. 1993.
Natarajan, N. and Slawsky, G. M. (1992). A Framework Architecture for Multimedia Information Networks. *IEEE Communications Magazine*. Feb. 1992.
Newcomb, S. (1991). The "HyTime" Hypermedia / Time-based Document Structuring Language. *Communications of the ACM*, December 1991.
Object Management Group (1991). *The Common Object Request Broker: Architecture and Specification.* OMG Doc. 91.12.1.
Parkes, A., and Self, J. (1990). Towards "Interactive Video": A Video-Based Intelligent Tutoring Environment. *Intelligent Tutoring Systems: At the Crossroads of Artificial Intelligence and Education* (Eds. C. Frasson and G. Gauthier). Ablex Publ. Corp.
Roca, R. (1985). ISDN Architecture. *AT&T Technical Journal*. 1985.
van Dam, A. (1987). Hypertext '87: Keynote Address. *Communications of the ACM*. July 1988.
Wallace, G. (1991). The JPEG Still Picture Compression Standard. *Communications of the ACM*. (34) 4.

The *Point and Query* Interface: Exploring Knowledge by Asking Questions

MARK C. LANGSTON and ARTHUR C. GRAESSER
Institute for Intelligent Systems
Department of Psychology, Memphis State University
Memphis, TN 38152, USA

There are substantial educational benefits in acquiring good question asking strategies. The learner improves comprehension and memory (King, 1989a, 1989b, 1991; Palinscar & Brown, 1984), achieves more self-regulated, active learning strategies (Collins, 1985; King, 1989a; Pressley et al., 1990), and accommodates transfer (King, 1991). Given such evidence, it would be beneficial to develop a pedagogical tool that enables students to ask good questions.

Research has shown that students are typically poor questioners; they ask very few questions in the classroom (Dillon, 1988; Graesser, Person, & Huber, 1992; van der Meij, 1988) and in individual tutoring settings (Graesser, Person, & Huber, 1992). However, it has also been demonstrated that once students have learned how and when to ask the right questions, there is a robust increase in both question frequency and sophistication (King, 1989; Palinscar & Brown, 1984). The only complication is that it takes many hours of explicit instruction to adequately train students in good question asking skills (King, 1989a, 1989b, 1991). The ideal tool would expose students to good questioning strategies and enable them to implicitly acquire and then apply these strategies.

The *Point and Query* Interface

We have developed a Point and Query (P&Q) Interface that facilitates question asking (Graesser, Langston, & Lang, 1991; Graesser, Langston, & Baggett, in press; Langston & Graesser, 1992). The P&Q interface is embedded in a hypertext environment and grounded in a psychological model of human question answering called QUEST (Graesser & Franklin, 1990; Graesser, Gordon, & Brainerd, 1992). Through use of the P&Q interface, students are exposed to different types of knowledge structures, and good questions associated with each structure. Long term integration of the P&Q interface into more traditional educational settings would allow students to apply learning goals developed in the classroom to their exploration of the knowledge contained in the P&Q system. Subsequently, they may apply the questioning strategies acquired during their exploration of the P&Q system to their interactions in the classroom.

The P&Q system differs from current computer based learning tools in several important ways, which are briefly summarized below. First, it is extremely easy for the student to ask a question. The student may ask a question through two clicks of a mouse button: first, to select an element from the screen about which to ask a question, and then the question to ask. This procedure takes about two seconds, as opposed to an average time of several minutes required to formulate a query in systems using a structured query languages, natural language interfaces, the Texas Instruments Menu Driven Natural Language Interface (Tennant, 1987), and RABBITT's "retrieval by reformulation" (Williams, 1984).

Second, it is extremely easy for the student to learn the use of the P&Q system. The average time to acquire navigation skills with the P&Q Interface is approximately 5 minutes. Training on other systems may require several hours. Third, the student has direct feedback regarding which questions are valid. When a student selects an element from the screen, the system responds with a list of questions relevant to that element, based on the QUEST model (Graesser & Franklin, 1990). In other interfaces, there is either limited feedback after the improper formulation of a question, or no feedback at all. Fourth, the student learns which questions are good questions. The questions presented in the menu are contingent on the type of knowledge structure associated with the chosen element, (e.g., goal/plan hierarchy, causal network, taxonomic hierarchy, or sensory/spatial information). The student will implicitly acquire knowledge about the relationships between knowledge structures and question types through extended use of the system. Fifth, the system is solidly grounded in a psychological model of question asking. Unlike other query systems, the P&Q system adapted a psychological model of human question answering as a guide for formulating answers to questions (Graesser & Franklin, 1990; Graesser, Gordon, & Brainerd, 1991). Sixth, the knowledge base is organized around questions and their answers. Schank (1986) discussed some of the advantages of organizing and indexing knowledge around questions, and there is some evidence that a "question plus answer" encoding format is particularly persistent memory, with slow forgetting rates (Pressley et al., 1988). Seventh, the P&Q system facilitates curiosity, active learning, and goal-driven learning. As will be discussed later, we have conducted several experiments to examine the feasibility of the P&Q Interface as an effective navigational system and pedagogical tool. We found that students asked approximately 127 questions per hour, 7 times the rate of question asking per student during normal tutoring and 800 times the rate of question asking per student in a classroom setting (Graesser, Person, & Huber, 1992). We also found a robust effect for learning goals on the types of questions asked, and therefore on navigational strategies. This suggests that the P&Q system is particularly applicable to goal-driven knowledge acquisition. Finally, the P&Q system contains a high concentration of relational links. This type of linking scheme is thought by some hypertext experts to be the essence of such a system, because of its relevance to performance and flexibility (Conklin, 1987).

Navigation using the P&Q Interface

The navigational approach taken in the P&Q system is a unique twist on the more typical means of navigation available to the user of a hypertext. The P&Q system allows each screen element, or node, to be connected to multiple links within the hypertext though the availability of multiple questions. In a typical session with the interface, the student selects an element he or she would like to know more about by pointing to it with the mouse and clicking the mouse button. The system responds by displaying a list of questions relevant to the chosen element, at which point the student may choose which question he or she would like to ask about the element. The system then presents the student with the answer to his or her element/question pair.

This navigational technique offers another advantage over standard educational hypertexts, which we term internodal coherence. Hypertext systems lend themselves easily to navigation through large, ill-defined knowledge domains, such as those described by Spiro and Jehng (1990). In such a domain coherence would be even more relevant than in closed, well-defined domains. However, the student is typically forced to infer the internodal coherence the author intended, because the student does not know what kind of information will be supplied when he/she activates a link. If the system supplies a choice of links at any one node, the link may not be explicit enough to convey to the student exactly what type of information the next node contains, in relation to the student's goals. The P&Q system of element choice/question selection overcomes this problem. By allowing the student to quickly formulate a question through a simple, two-step process, the student develops a coherent link between the next screen of information and the current screen. Until now the only systems capable of offering such internodal coherence have been slow and cumbersome to use. In addition, by supplying <u>questions</u> as the link determiners, the system can present the student with choices that closely match his/her formulated goals. The student also benefits from the psychological implications of knowledge organization around questions, as stated earlier.

Knowledge Organization in the P&Q Interface

The P&Q system presented in this paper is based on the semantic field of woodwind instruments. The knowledge is organized according to principles outlined by the QUEST model (Graesser & Franklin, 1990). The domain of woodwind instruments was chosen because of its ill-defined boundaries, and because it contains multiple levels of knowledge, each level having a tractable set of questions (see Table 1 below).

Questions associated with types of knowledge

Question	Theoretical Question Category	Type of knowledge
What does X mean?	Definition	Taxonomic hierarchy
What are the properties of X?	Feature specification	Taxonomic hierarchy
What are the types of X?	Concept completion	Taxonomic hierarchy
What does X look like?	Feature specification	Spatial composition
What does X sound like?	Feature specification	Sensory information
How does a person use/play X?	Instrumental/procedural	Procedural knowledge
How does X affect sound?	Causal consequence	Causal
How can a person create X?	Causal antecedent	Causal
What causes X?	Causal antecedent	Causal
What are consequences of X?	Causal consequence	Causal

Table 1.

The organization of knowledge in the P&Q system reflects not only a design issue, but also a push towards a more systematic, theory-based organization of hypertexts (Schank, 1986; Spiro et al., 1988; Spiro & Jehng, 1990). In contrast with Spiro and Jehng (1990), we believe that such hypertexts are a useful pedagogical tool for both the novice and the advanced learner. In addition, we feel that a well-organized hypertext coupled with the proper navigational tools can greatly facilitate the transition between the novice and expert levels of knowledge.

This paper presents two studies that investigate the feasibility of the P&Q interface as both a navigational system and a pedagogical tool. The first experiment investigated the correlation between the questions specified by the QUEST theory (see Table 1) and the questions freely generated by subjects presented with the same information. The hypothesis in this preliminary study was that there would be a substantial overlap between the questions used in the interface and questions students would normally generate given the same material. The second experiment examined the exploration patterns of subjects given various goals. We predicted that the learning goals of the subjects would have a substantial impact on the types of questions subjects asked over time, and therefore their exploration patterns. We also expected to observe a canonical ordering of knowledge exploration (see Dillon, 1984), where students would move from superficial knowledge to more task-specific or causal knowledge over time.

Experiment 1

This experiment examined the relationship between the questions contained in the P&Q interface (see Table 1), and the questions subjects freely generated when presented with information extracted from the interface. We expected to observe a strong overlap between the two sets of questions.

Method

Subjects

The subjects were 20 undergraduate students at Memphis State University who participated to fulfill a psychology course requirement.

Materials

The materials consisted a questionnaire to assess prior knowledge of musical instruments and two sets of packets. Each set contained 15 pages each of informational screens randomly extracted verbatim from the P&Q system. Both packets contained information from all knowledge types. In addition, each packet type contained two pages illustrating the two root nodes in the system: the woodwind taxonomy and a causal air flow diagram. There was a random ordering of information within each packet, and the information was identical within each set of packets.

Procedure

The subjects were screened to assess their prior knowledge about musical instruments, and were categorized as either novices or experts. Experts were rated as having moderate to high knowledge on a 6-point scale and also played a musical instrument. Novices were rated as having low knowledge on a 6-point scale and did not play a musical instrument. All subjects in this study were categorized as novices.

Subjects were randomly assigned to one of two experimental groups. Subjects in either group received one of the two packets according to group assignment. The subjects were instructed that the packets contained information about woodwind instruments, and that they were to treat the information as material to be learned. They were then instructed that they would have exactly 2.5 minutes to write down as many questions as they could generate about the information presented on a given page. At the end of the 2.5 minutes, the experimenter instructed them to turn the page and repeat the procedure for the new page of information.

Results

The questions generated by each student were categorized according to how well they mapped onto the questions presented in Table 1. Any question not immediately categorizable by the question types in Table 1 was categorized as type 'other', and not counted as a match.

The mean number of questions generated between groups was 362.5, with a mean of 222 categorizable questions generated. Therefore, 61.2% of the questions generated by the subjects were accounted for by the questions currently contained in the P&Q system. Most of the remaining questions were of the form "How does X relate to Y". This is a question type that the P&Q system cannot currently handle explicitly. However, the relationship could be inferred through multiple use of existing question types, such as "What are the properties of X?" followed by "What are the properties of Y?", for a relation between the properties of X and Y. Therefore, the questions contained in the P&Q system (see Table 1) are a valid set of questions, substantially overlapping the questions normally generated by students presented with the same information.

Experiment 2

Method

Subjects

The subjects were 48 undergraduate students at Memphis State University who participated to fulfill a psychology course requirement.

Materials

The equipment consisted of a questionnaire to assess prior knowledge of musical instruments identical to the one used in Experiment 1 and an instructional packet for the Point and

Query Interface containing three pages of instructions. Three types of task packets were used. *Design Instrument* packets required the subject to design a new woodwind instrument that was low in tone and pure in pitch. This task required a deep causal knowledge of woodwind instruments for task completion. *Assemble Band* packets required the subject to assemble a band consisting of six types of woodwind instruments, to perform at a New Year's Eve party. This task allowed the subject to rely on taxonomic, definitional, spatial and auditory information. *Free Exploration* packets contained the tasks described in both the Design Instrument and Assemble Band packets.

Procedure

We screened the subjects such that half the subjects had low knowledge about musical instruments and half had comparatively high knowledge. Those with high knowledge rated themselves as having moderate to high knowledge on a 6-point scale and also played a musical instrument.

We manipulated the goals of the learner so that one third of the subjects were expected to acquire deep causal knowledge of woodwind instruments, one third of the subjects were expected to acquire mostly superficial knowledge, and one third of the subjects were free to explore either superficial or causal knowledge. The Design Instrument condition required deep causal knowledge, the Assemble Band condition required only superficial knowledge, and the Free Exploration condition required both.

After screening, subjects were randomly assigned to one of the three experimental conditions, such that there was an orthogonal variation of prior music knowledge to learner goals. The subjects read the instructional packet and were given a 2 minute demonstration of how to use the P&Q system. The subjects were allowed approximately 1 minute to experiment with the P&Q system before they were given their tasks. Subjects in both the Design Instrument and Assemble Band conditions were told explicitly what their tasks were. Subjects in the Free Exploration condition were instructed to learn as much as they could about woodwind instruments during the interaction period, and that they would be tested at the end of the period. Subjects were allowed 30 minutes alone to interact with the P&Q system. At the end of the 30 minute interaction period the subjects were allowed as much time as necessary to complete the assigned task according to experimental condition.

Results

In an initial analysis, we computed the mean number of questions asked by subjects during the 30 minute interaction period. The subjects asked a mean of 75.6 questions per session in the Design Instrument condition, 59.9 questions in the Assemble Band condition, and 55.2 questions in the Free Exploration condition. Therefore, the rate of asking questions using the P&Q interface was 127.2 questions per student per hour. This is about 7 times the rate of question asking during normal tutoring sessions, and almost 800 times the rate of question asking in a normal classroom setting (Graesser, Person, & Huber, 1992). The P&Q software clearly facilitates questioning behaviour during learning.

The interaction period was segregated into three 10-minute time blocks, yielding blocks 1, 2, versus 3. The 10 question types in Table 1 were clustered into four knowledge categories: taxonomic-definitional, sensory, procedural, and causal. We computed the baserate percentage of questions in these four categories when considering all possible cards, content elements, and unique questions in the P&Q system. These proportions were 35%, 9%, 1%, and 55% for taxonomic-definitional, sensory, procedural, and causal, respectively.

An ANOVA was performed on question asking frequencies using a mixed design with four independent variables: condition (Design Instrument, Assemble Band, versus Free Exploration), prior knowledge (high versus low prior knowledge), time block (1, 2, versus 3), and knowledge type (taxonomic-definitional, sensory, procedural, and causal). Condition and prior experience were between-subjects variables whereas time block and knowledge type involved repeated measures variables.

The only significant impact of prior knowledge on questioning frequency was found in the Free Exploration condition. Subjects with high prior knowledge tended to ask more questions than those with low knowledge, with means of 55.38 and 44.88 questions respectively, $F(1,14) = 5.10$, $p < .05$. This outcome could be explained by the strong effect of the other experimental learning goals on questioning strategies. The constraints of the learner goals are extremely robust and tend to mask any prior knowledge effects.

The frequency of questions did not significantly vary as a function of time blocks, with means of 21.38, 21.88, and 20.10 questions in time blocks 1, 2, and 3, respectively. Therefore, the absolute volume of questions was approximately constant across the three 10-minute segments. The most questions were asked in the Design Instrument condition, followed by the Assemble Band and Free Exploration conditions, $F(2,42) = 4.87$, $p < .05$. The number of questions significantly differed among the four knowledge types, with means of 9.5, 5.1, 1.3, and 5.4 for taxonomic-definitional, sensory, procedural, and causal knowledge, respectively, $F(3,126) = 53.77$, $p < .05$. This outcome is not particularly surprising, due to the difference in baserate frequencies for each knowledge type.

There was a significant three-way interaction between condition, time block, and type of knowledge, $F(12,252)=2.72$, $p < .05$. Figure 1 plots the cell means that expose this three-way interaction.

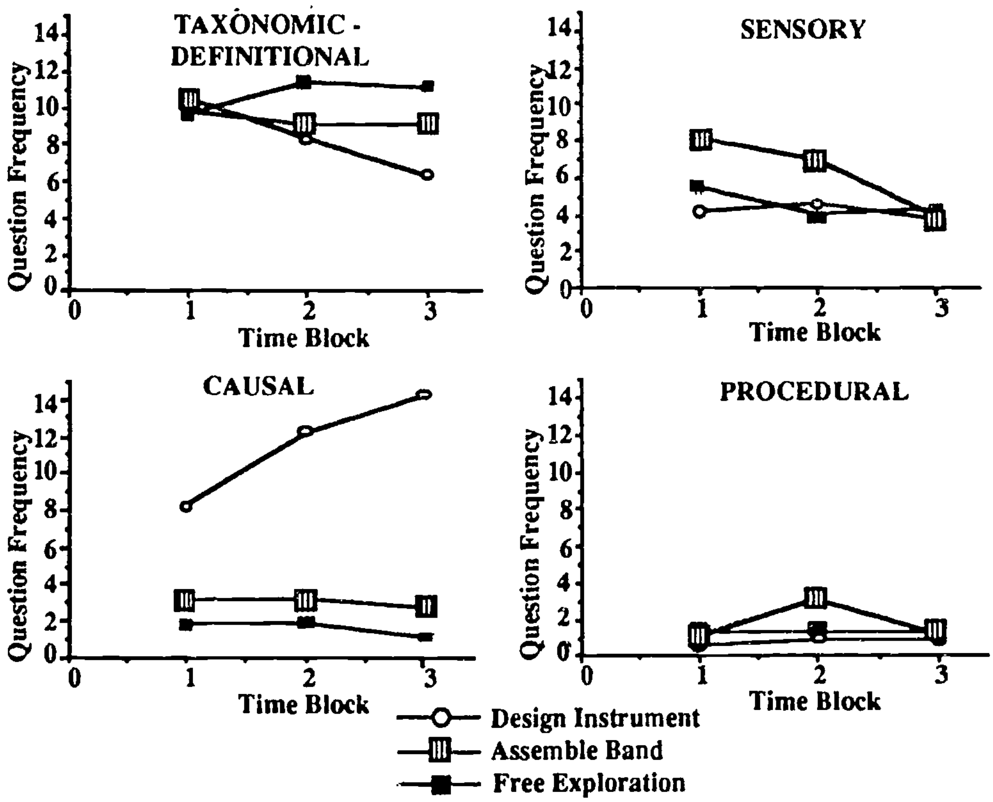


Figure 1.

Conclusions and discussion

We have documented how learners explore knowledge when they learn about woodwind instruments. Knowledge exploration was measured by observing the questions that college students asked about woodwind instruments during use of the Point and Query interface. We found that exploration patterns were unaffected by the students' prior knowledge of the domain under specific learning goals. In contrast, there were dramatic changes in exploration as a function of the goals and time course of the learning session.

The learners tended to sample taxonomic knowledge and definitions of terms during the first 10 minutes of the session. The learner's goals for each of the conditions were quite different: the Design Instrument condition required deep causal knowledge. Subjects in the Assemble Band condition could rely on superficial knowledge. Subjects in the Free Exploration condition were free to sample either type of knowledge in whatever order they chose. In spite of the differences in learner goals, the students needed to know about definitions of words and the taxonomic composition of the semantic field before exploring more goal-relevant knowledge. Thus, there appears to be evidence for a canonical ordering of knowledge exploration during the learning process (Dillon, 1984).

Once the taxonomic-definitional knowledge had been established, the student explored knowledge that more directly addressed learning goals. This is clearly supported when time blocks 2 and 3 are examined (see Figure 1). When deep causal knowledge was required, as in the Design Instrument condition, the learner explored this type of knowledge at the expense of the other types. In fact, causal knowledge was not explored unless the student was forced to seek this knowledge in pursuit of a goal. This is in direct contrast with information retrieval tasks, where the required information can be extracted from a very small number of screens (see e.g., Mohageg, 1992). In tasks such as these, the user tends to encounter navigational difficulties when the target information is distributed among several distant nodes. However, the subjects in our study did not appear to have any difficulty in retrieving the information necessary for task completion, even though the links were almost exclusively relational.

This study demonstrated the value of the P&Q Interface in investigating the process of knowledge exploration in hypertext systems. The P&Q Interface and other similar new interfaces (Schank et al., 1991; Sebrechts & Schwartz, 1991) have made it extremely easy for the student to ask questions. Until recently it has been very awkward and time-consuming to pose questions using navigational systems and this presented a serious barrier in investigations of information exploration. Interfaces such as the P&Q system present the opportunity to greatly enhance the efficiency and ease of use of various navigational systems for hypertexts, and to develop more pedagogically valid instructional hypertexts. It is possible that the P&Q Interface and systems like it could have a substantial impact on education, to the extent that they rekindle curiosity and promote good question asking skills. As discussed earlier, the learning of complex material in ill-structured domains can robustly improve if students learn how to ask the right questions (King, 1989a, 1989b, 1991; Palinscar & Brown, 1984). The P&Q Interface has the added potential to decrease search time and increase retention and recall.

References

Collins, A. (1985). Teaching and reasoning skills. In S. F. Chipman, J. W. Segal, & R. Glaser (Eds.), *Thinking and Learning skills (Vol. 2)*. Hillsdale, NJ: Erlbaum.

Conklin, J. (1987). *A survey of hypertext* (Report STP-356-86). Austin, TX: Microelectronics and Computer Technology Corp.

Dillon, J. T. (1984). The classification of research questions. *Review of Educational Research*, 54, 327-361.

Dillon, J. T. (1988). *Questioning and teaching: A manual of practice*. New York: Teachers College Press.

Graesser, A. C., & Franklin, S. P. (1990). QUEST: A cognitive model of question answering. *Discourse Processes*, 13, 279-303.
Graesser, A. C., Gordon, S. E., & Brainerd, L. E. (1992). QUEST: A model of question answering. *Computers and Mathematics with Applications*.
Graesser, A. C., Langston, M. C., & Baggett, W. B. (in press). Exploring information about concepts by asking questions. In G. V. Nakamura, R. M. Taraban, & D. Medin (Eds.), *Categorization by Humans and Machines*. San Diego: Academic Press.
Graesser, A. C., Langston, M. C., & Lang, K.L., (1991). Designing educational software around questioning. *Journal of Artificial Intelligence and Education*.
Graesser, A.C., Person, N., & Huber, J. (1992). Mechanisms that generate questions. In T. Lauer, E. Peacock, & A. Graesser (Eds.), *Questions and information systems*. NJ: Erlbaum.
King, A. (1989a). Effects of self-questioning training on college students' comprehension of lectures. *Contemporary Educational Psychology*, 14, 1-16.
King, A. (1989b). Verbal interaction and problem-solving within computer-assisted cooperative learning groups. *Journal of Educational Computing Research*, 5, 1-15.
King, A. (1991). Effects of training in strategic questioning on children's problem-solving performance. *Journal of Educational Psychology*, 83, 307-317.
Lang, K. L., Graesser, A. C., Dumais, S. T., & Kilman, D. (1992). Question asking in human-computer interfaces. In T. Lauer, E. Peacock, and A. C. Graesser (Eds.), *Questions and information systems*. Hillsdale, NJ: Erlbaum.
Langston, M. C., & Graesser, A. C. (1992). Question asking during learning with a point and query interface. In *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society* (pp. 921-926). Hillsdale, NJ: Erlbaum.
Mohageg, M. F. (1992). The influence of hypertext linking structures on the efficiency of information retrieval. *Human Factors*, 34, 351-367.
Palinscar, A. S., & Brown, A. L. (1984). Reciprocal teaching of comprehension-fostering and comprehension-monitoring activities. *Cognition and Instruction*, 1, 117-175.
Pressley, M., Symons, S., McDaniel, M. A., Snyder B. L., & Turnure, J.E. (1988). Elaborative interrogation facilities in the acquisition of confusing facts. *Journal of Educational Psychology*, 89, 301-342.
Pressley, M., Woloshyn, V., Lysynchuk, L. M., Martin, V., Wood, E., & Willoughby, T. (1990). A primer of research on cognitive strategy instruction: The important issues and how to address them. *Educational Psychology Review*, 2, 1-58.
Schank, R. C. (1986). *Explanation Patterns: Understanding mechanically and creatively*. Hillsdale, NJ: Erlbaum.
Schank, R., Ferguson, W., Birnbaum, L., Barger, J., & Greising, M. (1991). ASK TOM: an experimental interface for video case libraries. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society* (pp. 570-575), Hillsdale, NJ: Erlbaum.
Sebrechts, M. M., & Schwartz, M. L. (1991). Question-asking as a for novice computer skill acquisition. *Proceedings of the International Conference on Computer-Human Interaction*, 293-297.
Spiro, R. J., Coulson, R. L., Feltovich, P. J., & Anderson, D. K. (1988). Cognitive flexibility theory: Advanced knowledge acquisition in ill-structured domains. In V. Patel (Ed.), *Tenth Annual Conference of the Cognitive Science Society* (pp. 375-383). Hillsdale, NJ: Erlbaum.
Spiro, R. J., & Jehng, J. (1990). Cognitive flexibility and hypertext: Theory and technology for the nonlinear and multidimensional traversal of complex subject matter. In D. Nix and R. J. Spiro (Eds.), *Cognition, education, and multimedia* (pp. 163-205). Hillsdale, NJ: Erlbaum.
Sternberg, R. J. (1987). Questioning and intelligence. *Questioning Exchange*, 1, 11-13.
Tennant, H. R. (1987). Menu-based natural language. In S. C. & D. Eckroth (Eds.), *Encyclopedia of artificial intelligence*. New York: John Wiley & Sons.
Williams, M. D. (1984). What makes RABBITT run? *International Journal of Man-Machine Studies*, 21, 333-352.

FLAP: A Tool for Drawing and Simulating Automata

Mark LoSacco[+] and Susan H. Rodger[*]
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12180-3590

1 Introduction

In computer science, many of the fundamental concepts crucial to fully understanding the discipline have traditionally been presented abstractly, making it notationally difficult for undergraduates to comprehend. Animation provides an alternative way of presenting a concept. In the last few years, several tools have been developed that provide animations of some of these fundamental concepts. Balsa-II [2] and XTango [8] are systems for animating algorithms and data structures. There are also a number of systems for animating concepts in graph theory, sets, and combinatorics. Unfortunately, tools for studying the abstract concepts in automata theory are either textual tools or they are visual tools limited to studying one type of automata.

Automata theory provides the essential background for many areas of computer science, especially compiler design and programming languages. Traditionally, automata theory was taught at the senior level, usually as an optional course. Recently, its importance earlier in the curriculum was noted by the ACM Task force on the Core of Computer Science [4]. Many textbooks addressing automata theory and specifically designated for the sophomore level have been written in the past five years. Yet many students at this level have difficulty with the notation when automata are presented abstractly. We have developed a tool called FLAP (Formal Languages and Automata Package) that provides an interactive learning environment for students to design and experiment with automata. By being able to easily construct and then test their own designs, students can learn about automata in an enjoyable environment.

FLAP is a visual tool for designing, simulating and experimenting with several variations of finite automata (FA), pushdown automata (PDA) and Turing machines (TM).

* Supported in part by a Fellowship from the Lilly Foundation. Email: rodger@cs.rpi.edu

+ Current address is Bristol Technology, Inc., 241C Ethan Allen Highway, Ridgefield, CT 06877

ERIC

Students design an automaton by drawing it as a graph using a mouse. The graph representation is easier to visualize than the more traditional definition represented by a table. When the drawing of the automaton is complete, the student can simulate the automaton for some input. There is a choice of either a fast simulation or a slower step by step animation. In the slow simulation, a student can stop at any time and retrace all or part of the simulation. One of FLAP's features is it's ability to simulate nondeterministic automata. In this case, the fast simulation calculates all possible runs, and displays one solution. The slow simulation lets the user guide the simulation by eliminating certain possibilities when there are too many to consider. FLAP has been designed with a common interface so that once a student has studied one type of automata, it is easy to construct another type. In addition, the definitions of these machines are general so that several variations of each machine can be examined. This general definition will allow the tool to be used along with any automata theory textbook.

In Section 2 we discuss previous work on automata tools. In Section 3 we describe FLAP and its interface. In Section 4 we present an evaluation of using FLAP in teaching, and in Section 5 we describe future work.

2 Previous Work

Several systems for studying automata have been developed, but all of these systems either represent the automata in a tabular format, or they are specifically designed for studying only one type of automata. Sutner [10] developed a tool based on Mathematica for studying FA, and Dogrusoz [5] and Hannay [6] developed tools for experimenting with several types of automata, including finite automata, pushdown automata and Turing machines. However, none of these systems includes a graphical representation of the automata. In each of these tools the user must type in the formal definition of the machine. There are several tools that represent automata as graphs, but each tool only addresses one type of automata. Stallman [9] developed a graph tool which can be used to interactively draw an FA; AMS [7] is a tool for simulating FA; Turing's World [1] is a tool for simulating TM; and NPDA [3] is a tool for simulating nondeterministic PDA. FLAP is an expanded version of NPDA into a general tool for studying several types of automata. In addition, many new features were added to allow students more freedom in experimenting with their designs. These new features include additional tools to aid in designing an automaton, a fast simulation, and the ability to retrace part or all of a simulation.

3 Overview

FLAP is a graphical tool for drawing and simulating automata that was developed using C++, the X Window System and the Athena Widget set. In this section we first define the automata that FLAP can simulate, then we describe the drawing interface, the simulation interface, and mention other features.

The X Window System is a trademark of the Massachusetts Institute of Technology

3.1 Formal Definitions of Automata

FLAP recognizes a general definition of each type of automata so that several variations of each can be studied.

Definition 3.1: A nondeterministic finite automaton M is represented by a quint-tuple, $M = \{Q, \Sigma, \delta, q_0, F\}$, where Q is a finite set of states, Σ is a finite set of input symbols, δ is a set of transitions represented by $\delta : Q \times \Sigma^* \rightarrow 2^Q$, q_0 is the start state ($q_0 \in Q$), and F is a set of final states ($F \subseteq Q$).

The definition is general so that other variations of FA can be examined including deterministic FA, and restricting the number of input symbols that can be processed in a transition to one symbol. In all variations, a FA accepts an input string if there is a path from the start state to a final state that recognizes this input.

Definition 3.2: A nondeterministic pushdown automaton M is represented by a sept-tuple, $M = \{Q, \Sigma, \Gamma, \delta, q_0, Z, F\}$, where Q is a finite set of states, Σ is a finite set of tape symbols, Γ is a finite set of stack symbols, δ is a set of transitions represented by $\delta : Q \times \Sigma^* \times \Gamma^* \rightarrow$ subsets of $Q \times \Gamma^*$, q_0 is the start state ($q_0 \in Q$), Z is the start stack symbol ($Z \in \Gamma$), and $F \subseteq Q$ is a set of final states.

The definition is general so that variations of a PDA can be examined. Other variations include deterministic PDA, restricting the number of input symbols to one or zero (λ) symbols to be processed, and restricting the number of symbols popped on each transition to be exactly one. There are two definitions of the acceptance of an input string, providing additional variations of PDA that can be studied. Acceptance is based on either reaching a final state or the stack becoming empty.

Definition 3.3: A nondeterministic Turing machine M can be represented by a sept-tuple, $M = \{Q, \Sigma, \Gamma, \delta, B, q_0, F\}$, where Q is a finite set of states, Σ is the input alphabet, Γ is the tape alphabet (with $\Sigma \subseteq \Gamma - \{B\}$), δ is a set of transitions represented by $\delta : Q \times \Gamma \rightarrow$ subsets of $Q \times \Gamma \times \{R, L, S\}$, B is a special symbol denoting a blank on the tape, q_0 is the start state ($q_0 \in Q$), and $F \subseteq Q$ is a set of final states. The symbols R, L and S denote directions Right, Left, and Stay.

Other variations of the TM are a deterministic TM, and a variation when the movement of the tape head is restricted to always moving, either R or L. In all cases, acceptance of an input string is based on reaching a final state.

3.2 Drawing an Automaton

The *Building Window* provides the means to graphically design an automaton. In addition to containing the canvas on which the automaton is drawn, this window houses a variety of pull-down menu buttons, the input tape, and the acceptance criteria indicator. The Building Windows for a PDA and a TM are shown in Figures 1 and 2 respectively. The PDA of Figure 1 non-deterministically accepts the language, $L = \{ww^R \mid w \in \{a, b\}^+\}$, while the deterministic TM of Figure 2 accepts the language, $L = \{a^n b^n c^n \mid n \geq 1\}$.

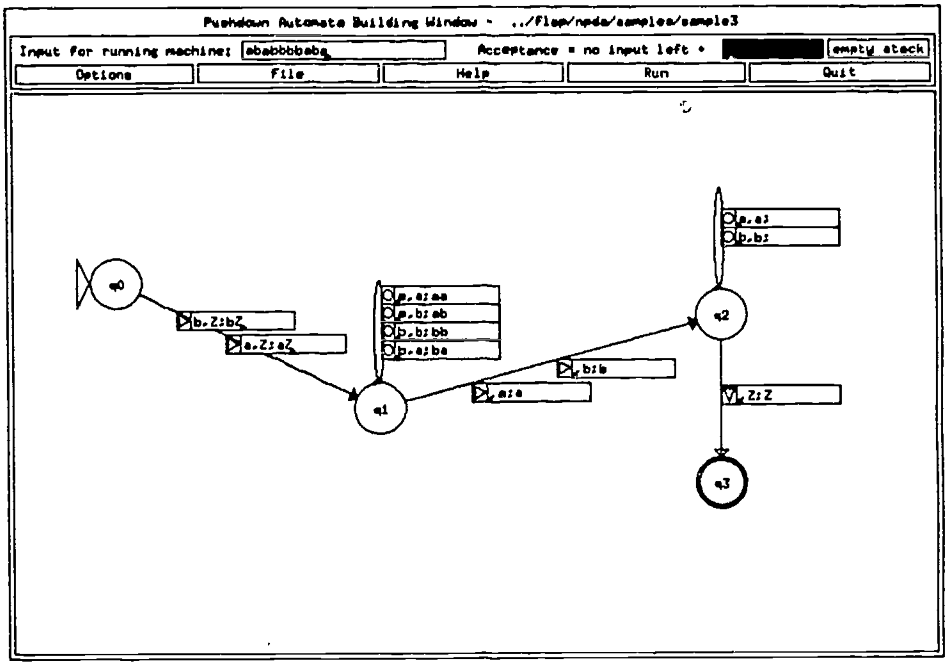


Figure 1: Pushdown Automata Building Window

States, which are created by clicking the mouse on the canvas, are depicted as labeled circles. The initial state is drawn with a triangle pointing to it, while final states are drawn as concentric circles. The initial state of the automaton of Figure 1 is q0, while the final state is q3. An existing state may be deleted, moved about on the canvas, made final or non-final, and designated as the initial state by positioning the mouse within it and using the appropriate mouse button.

Transitions are represented as labeled directed arcs which connect two states. When multiple transitions exist between the same two states, as in the transition from state q0 to q1 of Figure 1, the transition labels share the same line. Transition labels consist of two parts: a handle and a text field. The handle is used in moving the label and deleting the transition. Within each handle is a graphic symbol (◁, ▷, △, ▽, ◯) which shows the direction of the transition. The text field to the right of the handle holds the transition information.

Adding transitions to a machine involves positioning the mouse within the origin state, and holding down the mouse button while moving into the destination state. The transition is created when the button is released. A loop transition, which results from the origin and destination being the same state, is created by clicking the mouse button within the state. A loop transition is denoted by a ◯ in the transition label handle. Newly created transitions have blank text fields, which may be edited by positioning the cursor within them and using the keyboard to edit the contents. The format of the transition label information is machine model dependent, and therefore different for each of the tools.

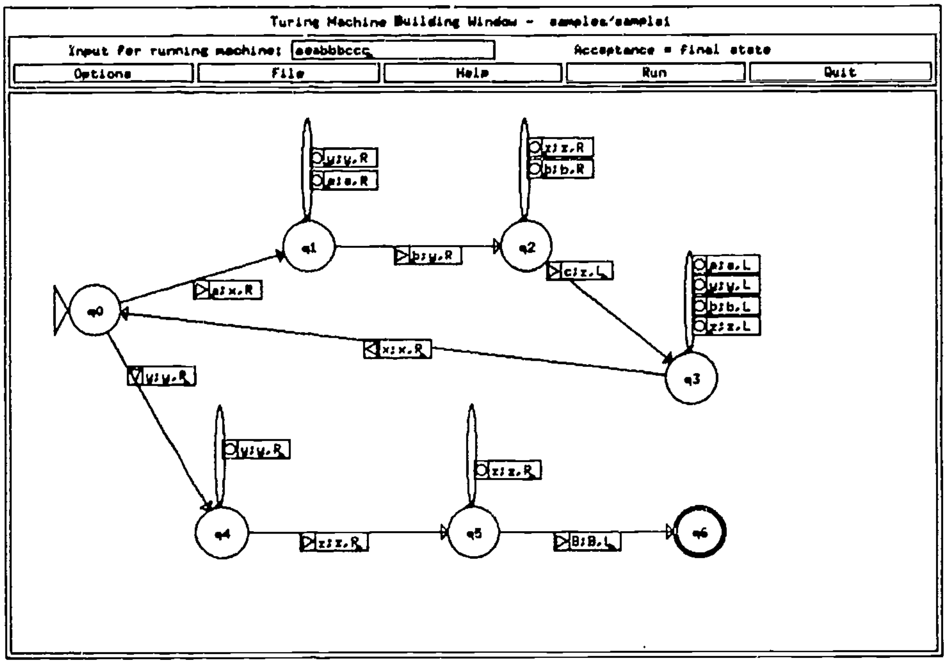


Figure 2: Turing Machine Building Window

The transition format for all machines is:

FA transition format:	$<$ *in_string* $>$
PDA transition format:	$<$ *in_string* $>$, $<$ *pop_string* $>$; $<$ *push_string* $>$
TM transition format:	$<$ *in_symbol* $>$; $<$ *write_symbol* $>$, $<$ *dir_symbol* $>$

where $<$ *in_string* $>$ is the (zero, one or more) symbols that must be present on the input tape if the transition is to be applied, $<$ *pop_string* $>$ is the symbols that must be present at the top of the stack which are to be popped off, $<$ *push_string* $>$ is the symbols that are to be pushed on the stack, $<$ *in_symbol* $>$ is a single symbol which must be the current symbol on the input tape if the transition is to be taken; $<$ *write_symbol* $>$ is the single symbol which is written to the current position on the input tape; and $<$ *dir_symbol* $>\in \{R, L, S\}$, depending on whether the read head should move Right, move Left, or Stay put. For $<$ *pop_string* $>$ and $<$ *push_string* $>$, the leftmost symbol corresponds to the top of the stack. The symbol recognized by PDA as the start stack symbol is Z. The symbol recognized by TM as the blank symbol is B.

The acceptance criteria, which is displayed in the upper right corner of the Building Window, is also machine model dependent. In the case of FA, the acceptance criteria is *No Input Left + Final State*, signifying that the machine must be in a final state once the entire string has been processed in order to accept. The criteria for acceptance by PDA is either *No Input Left + Final State*, or *No Input Left + Empty Stack*. The currently selected criteria is displayed in inverse video. Figure 1 shows a PDA that accepts by final state. A PDA that accepts by empty stack requires that the entire input be processed. and the stack

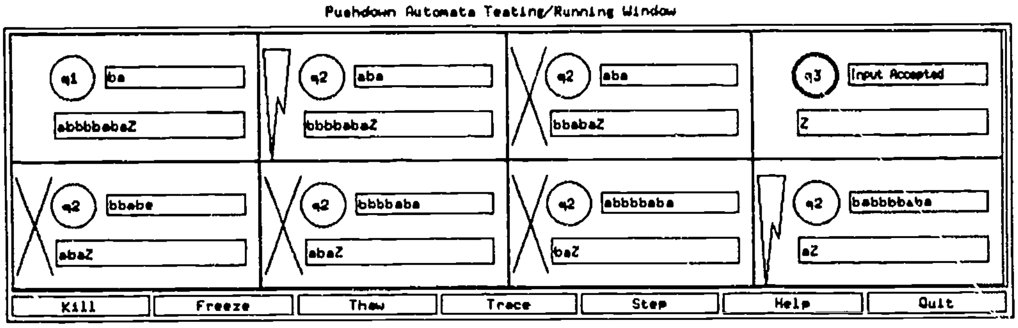


Figure 3: The Run Window

be empty. Lastly, the criteria for acceptance by TM is simply *Final State*, signifying that the machine accepts whenever a final state is reached.

The input string is specified by positioning the cursor within the text entry box located in the upper left portion of the Building Window, and using the keyboard to edit the contents. The input string may be changed at any time during which the Building Window is active. The input strings in Figures 1 and 2 are *ababbbbaba* and *aaabbbccc* respectively.

3.3 Simulation of Automata

Once a machine has been created or retrieved, it may be simulated on some input by selecting *Step Run* or *Fast Run* from the Run menu in the Building Window. Step Run allows the user to *step* through the execution of the machine, viewing the current configurations of the machine along the way in a separate window called the *Run Window*. Figure 3 shows the Run Window during a run of the PDA of Figure 1. The input string *ababbbbaba* was accepted in state q3, after 12 steps.

A configuration is a snapshot of the current situation, containing only the information needed by the machine to continue from that situation. The format of a configuration is machine model dependent. The FA configuration consists of the current state and the remaining input string. A PDA configuration includes the same information as that of the FA, but adds the current stack contents, displayed below the state. Lastly, the TM configuration format consists of the current state and the input tape. The current position of the read head on the tape is centered within the tape, pointed to by arrows from above and below.

Upon each application of the Step command, the current set of configurations are expanded by the application of all possible transitions to create a new set of configurations. Configurations which are marked by X's, such as four of the configurations in Figure 3, are stalled and may no longer be expanded. A maximum of 12 configurations may be kept in the current set at any given time. The current states of the machine, which correspond to the current set of configurations, are shown highlighted in the Building Window during the simulation, so the user can easily observe the paths taken through the automaton.

ERIC Full Text Provided by ERIC

While the Step Run is very useful in debugging or analyzing an automaton, there are times when a faster method of testing a machine is desirable. The Fast Run, which simulates a machine without the use of the Run Window or user intervention, is provided for this purpose. The result of the simulation is displayed in a popup window when the input is accepted, rejected, or a cutoff point is reached. The cutoff points, which are also enforced by Step Run, guard against the possibility of the machine running infinitely.

3.4 Other Features

In addition to the building and running features, there are several other options and features provided by the tools. Input and output features include the ability to save, retrieve, and print machines. A *Check Determinism* operation provides a means to determine whether the current machine is deterministic or non-deterministic, in which case the non-deterministic states are highlighted. States may be re-labeled automatically or on command to keep the numbers consecutive. Lastly, on-line help is provided for quick reference during the use of the tools.

4 Evaluation

FLAP is used in the sophomore class sequence Fundamental Structures of Computer Science I and II at Rensselaer. Approximately half of each of these classes focus on automata theory and formal languages. The automata studied are FA, PDA, and TM, in this order. When a new automaton is studied, students are given an assignment to design a specific automaton, and a one page set of instructions to get started. The students use FLAP to design the automaton, and then use the simulation and tracing features to verify its correctness. When they are satisfied that the automaton is correct, they can easily print a copy of it to turn in.

Here is an example of an assignment for pushdown automata.

- Construct nondeterministic pushdown automata for the following languages.

    1. $L = \{a^{n+m}b^nc^m | n \geq 0, m \geq 0\}$. Acceptance should be by final state.
    2. $L = \{a^nb^m | n > 0, m > 2n\}$. Acceptance should be by empty stack.

As part of the assignments, students are asked to evaluate the tool and list any problems they had in using it. The majority of their comments have been positive with many students saying that the most helpful part of the tool was the ability to simulate their designs. Overall, we have seen more enthusiasm for learning automata theory than in previous semesters when students wrote similar assignments using pencil and paper, not being able to test out their designs.

5 Future Work

FLAP should represent a package of tools for studying formal languages and automata. Currently it only has a few tools for studying automata. Its design includes an initial menu to select one of the three automata: FA, PDA, or TM. This design makes it easy to incorporate additional tools into the package. Future work includes adding tools for studying grammars, parsing, additional variations of Turing machines, and construction methods (such as converting a nondeterministic FA into an equivalent deterministic FA and then constructing a minimum state FA).

Acknowledgements Dan Caugherty wrote the prototype NPDA which FLAP is based on. In addition, the authors wish to thank Matt Stallman and numerous Rensselaer students who gave valuable comments for improving FLAP.

References

[1] J. Barwise and J. Etchemendy, Turing's World, Kinko's Academic Courseware Exchange, Santa Barbara, CA, 1986.

[2] M. Brown, Exploring Algorithms using Balsa-II, *Computer*, Vol. 21, No. 2, p. 14-36, May 1988.

[3] D. Caugherty and S. H. Rodger, Npda: An Application for Building and Testing Pushdown Automata, User Manual, Rensselaer Technical Report 92-7, Feb 1992.

[4] P. J. Denning et. al, Computing as a Discipline, In *Communications of the ACM*, Vol. 32, No. 2, pp.9-23, Jan. 1989.

[5] U. Dogrusoz, Abstract Formal Machine Simulator, Computer Science Department Master's Project, Rensselaer Polytechnic Institute, 1991.

[6] D. Hannay, Hypercard Automata Simulation: Finite State, Pushdown and Turing Machines, *SIGCSE Bulletin*, 24, 2, p. 55-58, June 1992.

[7] M. C. Lee, An Abstract Machine Simulator, *Third International Conference on Computer Assisted Learning*, p. 129-141, 1990.

[8] J. Stasko, Tango: A Framework and System for Algorithm Animation, *IEEE Computer*, p.27-39, Sept. 1990.

[9] M. Stallman, R. Cleaveland, and P. Hebbar, GDR: A Visualization Tool for Graph Algorithms, North Carolina State University Technical Report TR-91-27, (1991).

[10] K. Sutner, Implementing Finite State Machines, *DIMACS Workshop on Computational Support for Discrete Mathematics*, 1992.

MULTILINGUAL CBT

Douglas MacKenzie
Managing Director, DMC Ltd
The Towers, Woodside House, Arbroath DD11 3RB, SCOTLAND

The Need and the Problems

In recent years the need for CBT courses on the same topic to be developed in a variety of different languages has grown considerably. The development of the Single European Market with transnational companies, the simultaneous introduction of computer systems, new products and procedures in the range of countries where an organisation operates and the demand for training in the emerging democracies of Eastern and Central Europe have all created large training needs. Multimedia training is ideal for teaching large numbers of people in geographically dispersed sites but many problems present themselves in producing versions of a single course in several languages.

The little information on this topic to be found in the literature (e.g. Trollip and Brown, 1987) tends to describe the translation of existing material from one language to another. In commercial training there may not be time for this: the Madrid office requires the training at the same time as the Glasgow office. Development time, therefore, is a factor as is, of course, cost. In software product development, the cost of producing another country's version from an original may be close to the original development cost. This is acceptable for large volume sales but not for bespoke development of CBT where the target audience in one particular country may be very small but, nevertheless, needs training in its mother tongue. In software development the audience needs to adapt to the product: Windows 3.1 looks the same the world over. A training course, on the other hand, must adapt to the needs of the audience. The demands on the foreign language course developers are great: they must understand both the subject matter of the course and the principles of CBT.

Technically there are problems to solve with character display and recognition. A language's characters may not be represented in ASCII. Even where they are there are problems. The letter ü in German is usually written as ue in the German-speaking parts of Switzerland. Answer analysis needs to take into account the fact that some users will have a German language keyboard, others an English one and that u, ü and ue may all be input to mean the same thing. The problem becomes greater with characters such as the Polish ł where no ASCII representation exists. A language like Arabic where script is from right to left but numbers are written from left to right poses great difficulties in answer fields where input may be alphabetic, numeric or mixed.

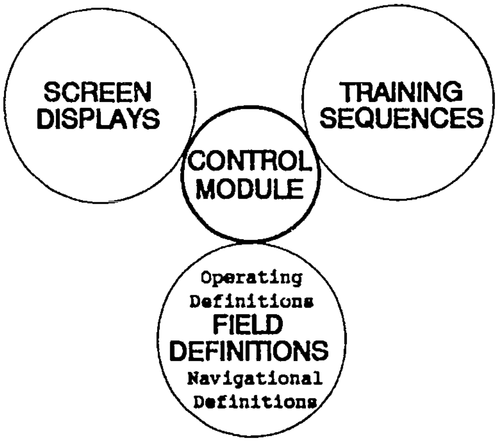
The Project

Developers at DMC had been involved in producing CBT courses in German and French and in producing a Polish version of a banking course first developed in English so some experience of design and development problems had already been garnered. The first major application of this experience was the production a course on an aspect of credit-card processing commissioned by VISA International to be delivered in English, French, German, Italian and Spanish. The initial development was to be done on an English version of the course. The other language versions were to follow as quickly as possible. The design brief restricted the delivery machines to the lowest common denominator: IBM PC compatibles equipped with EGA graphic cards. The speed with which the multiple versions were required and cost constraints (the different language versions were each budgeted at a maximum of 15% of the cost of the English version) conditioned the development strategy from the start.

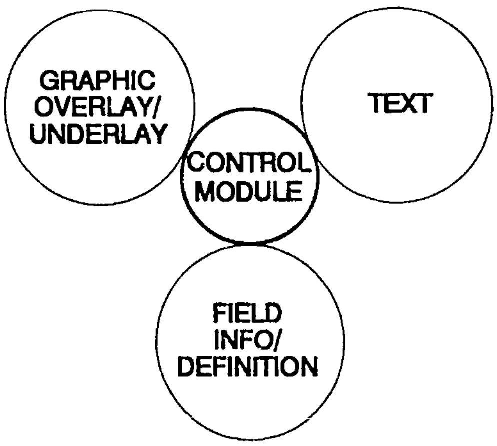
The Development Methodology

All CBT development projects at DMC use a methodology known as instructional prototyping (MacKenzie 1987, 1990). This approach, unlike ISD, does not separate design from development. Prototypes of the course exist in a computer-based form from the start of any project and, acting as templates for development, replace both initial and detailed design documentation. In this way, a version of the course always exists on computer for all subject matter experts, trainers and other interested parties to review.

An extension of this, part method, part software toolkit, called CIRCUS (MacKenzie, 1991) has been used with great success in creating courses which involve the simulation of computer systems. As CBT development has often to track development of a live computer system, CIRCUS is an attempt to reduce the amount of changes to a CBT package necessitated by alterations in the 'real' system. This is achieved by dividing course material into three parts: the appearance of the computer screens (a 'data dictionary' of sorts), the navigation between these screens, and the teaching material. The three parts are under the control of a central module. The idea is that a change to one screen in a real computer system will require only one change in the CBT screen dictionary and one change in the definition of fields in the navigation module. The trainer responsible for writing the material then only needs to amend the text in the teaching material section. Our earlier experiments suggested this was a good model for multilingual course development also. The intention was to have between 50 and 70% of the code constant in all versions with translators needing to work only within the text module. The relationship between CIRCUS for Information Technology training and its use in this project is shown in Figure 1.



In training end-users of computer systems



For multilingual CBT

The course was written in PC/Pilot for both commercial and technical reasons: commercially, because no run-time licences are involved and the customer wished to distribute the course to an unknown number of end-users. Technically the language was chosen as much of the CIRCUS toolkit had been implemented in PC/Pilot because of the sophistication of the language in setting up conditional subroutine calls, an integral part of the communication between the control module and the other three.

The First Mistake

The adoption of the CIRCUS approach and instructional prototyping was certainly the first thing we did right. It allowed core routines to be developed for all language versions and, once an English version of any course section was available, it allowed translator/developers to work on their language version of that section independently. This power and flexibility, however, led directly to the first thing we got wrong. Trying to develop everything simultaneously resulted in too many changes being made at one time. In response to both internal and external reviews of course prototypes (there were four formal prototypes during development and three ad hoc ones) changes were being made as a result of requests in four distinct areas.

> The identification of genuine bugs where the program did not operate as expected.
>
> Content changes requested by subject matter experts.
>
> Re-phrasing of text either reflecting technical content or preferred terminology in the different target languages.
>
> Extension of the answer analysis done in the course due to new anticipated wrong answers being identified and, again, country-specific (rather than language-specific) terminology being added.

Although the objective of keeping 70% of the code generic was achieved, the effort of maintaining the remaining 30% for five different languages under development with the four different types of change request constantly coming in was underestimated. We found no commercially available version management package to meet our needs exactly. As a result too much time was spent collating and managing code. This effort could have been reduced somewhat by delaying work on other language versions until after all the comments from the beta-test sites had come in but, when several versions are to be produced almost simultaneously, this problem is always going to arise. There will always be some country-specific or language-specific changes identified late in the development cycle which must be allowed for in the timetable and budget.

The Second Mistake

The types of interaction in the course were varied: highly graphical tutorial sequences; form filling exercises; question and answer sequences; on-line reference material and case studies where the documents and systems available in the workplace were accessed by pressing function keys. There were, therefore, no definite screen design templates in the course. However, as many parts of the course involved representations of computer reports or other documentation, the general approach was to depict the documentation as accurately as possible and show the teaching material as distinct from this by using a dialogue box. We have definite ideas as to how much a trainee is prepared to read in a dialogue box: about five lines of text with no more than 25 characters per line. In line with the intention of keeping the program code constant throughout all versions we sought to use a standard sized dialogue box displayed and updated by the same routines in every version. Although it lays us open to the criticism of allowing the medium to dictate the message, we have found the discipline of a specified size of dialogue box helps developers (and especially reviewers with little CBT experience) crystallise their ideas so that an explanation fits the box and the trainee's perceived attention span.

So, based on our experience of translating paper-based materials we anticipated German to be the most verbose version we would produce, requiring around 30% more space than the English text. To allow for long compound nouns in German which we did not wish to hyphenate between lines, we added 35% to the size of the basic English-version dialogue box and used this as our standard.

Unfortunately, in reviewing the prototypes, the discipline vanished. Reviewers looked at the English text in a sparsely filled dialogue box and saw that more text could be added within the space. The changes requested meant that the dialogue box which was a half to two-thirds full was now three-quarters to totally full in the English version. This made the translation, particularly into German, more difficult and required the developers, on occasion, to use a terser style of language than was desired.

The Third Mistake

The CIRCUS development method was based on our belief that it was far better for the foreign language developers to work directly with the program (or the text files linked to it) rather than translate on paper or even via ASCII files. This belief arose from our commitment to prototyping (Mackenzie, 1987) and experience of designers working with coding sheets: if forty spaces are marked on the sheet corresponding to screen positions the designer invariably solves his problem of surplus text by writing two characters in one space! Although layout is nowadays far more flexible the basic problems are the same. So it proved. The third mistake was even to contemplate allowing development to be done away from a machine. The French and German versions were developed entirely at the terminal with the only problems arising from the second mistake described above, allowing the English text to become too long. The Italian and Spanish versions had portions developed remotely and this resulted in passages of text not fitting into their allocated space, hanging over the edge of dialogue boxes or feedback being split over two lines. This latter problem was particularly difficult to identify in the testing phase as there were over 600 separate pieces of feedback used in the Case Studies part of the course. The improved layout facilities in modern courseware magnify the problem: the length text will be when printed with proportional spacing and in different fonts is very difficult for a translator to judge.

Eventually we had all developers working directly with the code and testing as they translated. This was the second and most important thing we got right and we will use it as a model for all future development of this nature.

The Fourth Mistake

The ability to use different fonts alluded to above led to a bizarre mistake. In our earlier foreign language courses we had used only user-defined typefaces, usually 10, 12 and 14 point Helvetica fonts. This was for aesthetic reasons in German and practical ones in Polish in that the ASCII character set could not represent all the characters needed. For no particularly good reason we used restricted 128 character fonts and redefined characters we did not need, such as *, to be the foreign language character required such as ê. In the VISA course we used a mixture of user-defined 12 point Helvetica fonts where characters had to be redefined and system fonts as, for the languages used, all the characters were available in the ASCII character set. This meant that translators had sometimes to input *, sometimes ê, to represent the same character. This caused a deal of confusion for live input and absolute havoc for files transmitted to and fro in ASCII format resulting most memorably in a Spanish translator being sent some files to review and spending an entire weekend removing the user-defined characters believing them to be corruptions.

In this project, with hindsight, we should have used 256 character user-defined fonts so that ASCII codes always corresponded whatever font was being used. This is not, however, a general solution as it cannot apply to Cyrillic, Turkish or other non-Western European alphabets which use non-ASCII characters. In these cases extra thought must be given to how input and editing are to be handled when multiple language versions are being produced.

The Fifth Mistake

In developing the five language versions we assumed we could apply what we thought we knew about CBT to all the courses. Keep explanations simple; break procedures down into small steps; use frequent summaries with numbered items to aid recall are fairly standard guidelines in all our courses. The early prototypes of the VISA course revealed that this approach was not universally popular: reviewers in one country felt it made the material appear too simple; the trainer responsible for another language version disliked 'disjointed text' and wanted material to read like a book. The fact that we had asked trainees to enter their name (mainly for record-keeping purposes) and then personalised a graphic of a credit card with that name was considered inappropriate in one country particularly if the full title of the trainee was not displayed.

Despite the mistakes, the VISA courses can be judged a success. They are in use throughout Europe, the Middle East and Africa and have also been bought by banks in Latin America and Asia. The project won the 1991 Philips IMS 'Vision in Interactive Media' Award. Late nights resolved the technical problems. The instructional prototyping methodology which identified the 'cultural problems' early on meant that these could be remedied without major difficulties even if sometimes the course became a little more bland.

General Conclusions

It is, however, these cultural differences which have proved the most awkward in all the multilingual CBT we have developed and which involve the most far-reaching design considerations. Much of the CBT we develop involves teaching by analogy and in doing this we make extensive use of graphics and animation. Trying to find examples which span languages and cultures can be difficult. An example in a finance course of a banker 'wearing many hats' for his different roles was illustrated by an animation of Lewis Carroll's Mad Hatter. Unfortunately, in producing the Polish version of this course, we learnt that the appropriate Polish metaphor was 'wearing many gloves'. In another course, some users were puzzled why savings were being illustrated by coins disappearing into a pig's back.

Paradoxically, the more technically complex the course delivery process, the easier it is to resolve the cultural differences. If audio is available the constraints of space for text on screen no longer apply. Similarly, if still or moving video images are used, real-life situations can be depicted without worrying whether a graphic or iconic representation will be universally understood. The use of software only JPEG and MPEG compression for audio and video has been of enormous benefit to us in creating multilingual courses. However, many multilingual courses will continue to be commissioned in order to bring less well developed regions up to speed on particular topics and a lowest common denominator approach to equipment will often have to be taken. Given the constraints of, say, EGA graphics, developers will have problems of cultural differences to resolve. There are no rules or simple guidelines for design in these cases but it is hoped that our experiences will help others at least avoid the straightforwardly technical problems and allow them to concentrate more on content.

References

MacKenzie, D. (1987). Instructional Prototyping: A CBT Development Strategy, Proceedings of the International Conference on Computer Assisted Learning, University of Calgary, 157-163.

MacKenzie, D. (1990). A human approach to computer-based course development: instructional prototyping. In Farmer, B., Eastcott, D. and Lantz, B. (eds), Aspects of Educational and Training Technology XXIII, Nichols, N.Y., N.Y., 220-225.

MacKenzie, D. (1991). Developing CBT Courses for End-User Training, Proceedings 33rd ADCIS Conference, 345-351.

Trollip, S.R. and Brown, G. (1987). Designing Software for Easy Translation into Other Languages, Journal of Computer-Based Instruction, 14(3), 119-123.

MPC: An Evolving Standard in Multimedia Education

YOGESH MALHOTRA
Department of Management
University of Nevada at Las Vegas, NV 89154, USA

RANEL ERICKSON
Department of Management
University of Nevada at Las Vegas, NV 89154, USA

For several years, multimedia capabilities have been available on diverse hardware. The current contenders in the desktop multimedia marketplace include the PC-compatibles, IBM PS/2, Apple Macintosh, Commodore Amiga, NeXTstation Color, and Silicon Graphics' IRIS Indigo. We believe that Microsoft's MPC standard, a relatively new entrant in the multimedia marketplace, will play a significant role in the evolution of a de facto industry standard for multimedia education.

Amiga, which represented the first true multimedia computer, has been used for creating desktop TV productions because of its special video capabilities. The major US multimedia product developers do not support Amiga because of its low installed base (of approximately 3 million units), offbeat array of graphic modes, non-standard 24-bit graphics, the smallest out-of-the-box color palette, and comparatively low resolution.

Apple's Macintosh, the desktop preferred by artists, publishers, and musicians, has since long combined digitized sound, still or animated images, and music or video from an external CD or laser disc. Apple's QuickTime extension of its operating system provides for new dynamic data types that can contain audio and video among other data types. Presently, Apple is working on the development of its new Kaleida technology in collaboration with IBM.

The recent advances in the PC-compatible market, such as Windows 3.1, MPC, and color resolution, have provided for standards and capabilities that have resulted in powerful software/hardware solutions for authoring, animation, graphics, and sound. The price/performance ratios for this market segment have been the most favorable because of the relatively large market share of these machines.

IBM is backing its "Ultimedia" family of multimedia products built around its PS/2. It is focusing on providing complete turnkey multimedia development and delivery solutions.

NeXT, although a powerful machine, has not yet been widely accepted because of the lack of software support and because of being an unproven standard. Presently, its use is limited to multimedia production installations, one-time shows, and video production.

The IRIS Indigo possesses unique high-end graphic capabilities that have been used for creating special effects like those in the movie *Terminator 2*. Indigo is a strong candidate for video and multimedia production, as well as for one-time presentations. Philips CD-I (Compact Disk Interactive), Commodore's CD-TV, and the NEC Turbografx, represent examples of VCR-like multimedia CD players that can be hooked to a TV set. Most of these products are not compatible with each other.

What is Multimedia Personal Computer (MPC)?

On October 8, 1991, Microsoft started a new era in PC computing by introducing the MPC or Multimedia Personal Computer standard. The MPC provides a standard that brings multimedia capabilities to PCs and compatibles. To be a Multimedia PC (or an MPC), a system should satisfy the standards laid down by the Multimedia PC Marketing Council, a subsidiary of the Software Publishers Association. The goal of the council, which is comprised of Microsoft and a growing group of hardware manufacturers, is to provide a common standard for multimedia computing on the PCs. To the developers, MPC is now a standard for creating advanced multimedia applications. To the users, MPC provides the specifications for buying new systems and upgrading the existing ones to support those applications. MPC is a comprehensive standard and provides for affordable stand-alone systems as well as upgrade kits. The MPC standard is described in detail in Microsoft's *Multimedia PC Specification Version 1.0*. To meet the MPC standard, a system should satisfy the following requirements:

i) it should be capable of running Microsoft Windows 3.1 or Microsoft Windows 3.0 with Multimedia Extensions;
ii) it should have VGA or super-VGA graphics;
iii) it should have a relatively fast CD-ROM drive; and
iv) it should possess specific high-end sound capabilities.

The MPC council's minimum specifications for a MPC include an Intel 10-MHZ 80286 CPU; 2 MB of system memory (RAM); VGA graphics; an audio card with 8-bit sampling, music synthesizer, MIDI-in/MIDI-out and on-board analog audio mixing; a CD-ROM drive with CD-Digital Audio (Red Book) output; MS-DOS 5.0, and Microsoft Windows 3.0 with Multimedia Extensions. The key elements of the specification are detailed in Table 1.

All multimedia systems produced by MPC Marketing Council, or any of its licensees, would bear the MPC logo. Systems that conform to the MPC standard, but are not manufactured by any of the previously mentioned companies, would be referred as MPC-compatibles.

Multimedia, Education and the MPC

Multimedia is about more than charts, graphs, and 'flat' art. It's about mixing text, images, voice, music, sound effects, animation, and video. For communication of any kind to be effective, it must draw upon the full power of how we experience life -- and that means multimedia communication that presents information in its most naturally expressive forms. Because of the added 'dimensions,' the multimedia applications are called 'titles' to distinguish them from 'traditional' applications.

Making learning enjoyable (a process of having fun) is a time-tested stratagem that has gained new appeal with the advent of the Multimedia PC. Children, as young as four, can now learn languages through interactive lessons that feature voices and images of native speakers simulating real-life situations. The multimedia education offers students and teachers greater flexibility in addressing individual needs and pursuing personal interests as well as the standard curricula. The process of developing research skills and learning becomes a process that provides pleasure and fun.

Multimedia Windows and Multimedia Extensions

The Multimedia Extensions are an addition to Microsoft Windows system software. They enable PCs running Windows to use multimedia devices and data types, such as sound, animation, video and CD-ROM capabilities. Features include the Media Control Interface, which enables Windows users to control external devices like audio and video recorders and players.

Table 1
Key Elements of the MPC Standard

	Official 'Minimal'	Desirable (where applicable)
Microsoft Windows		
CPU	10Mhz 286	At least a 386SX
RAM	2 MB	4 MB or more
Hard Disk	30-MB	More than 100 MB for creating graphics & sound
Video	Standard VGA	VGA+
CD-ROM Drive		
Storage	600MB	More
Data Transfer	150KB per sec	More
CPU Access Time	40% of total	Less
Access Time	1 second	Less
Plays CD audio		
Audio		
Sound Board		
Sampling	8-bit, 11.025-Khz	
Playback	22.05-Khz	
Synthesizer		
	8 simultaneous notes	
	5 different sounds	
Mixer (combines)		
	CD audio	
	CD sampling	
	Synth	
Microphone input		
MIDI		
	Input port	
	Output port	
	Thru port	
Floppy Drive	3.5 in 1.44 MB	
Keyboard	101 key	
Mouse	2-button	
Ports		
	Parallel	
	Serial	
	Joystick	

Originally, the Windows software with the extensions loaded on top of it was known as *Microsoft Windows 3.0a with Multimedia Extensions 1.0.* In early 1992, Microsoft launched *Windows 3.1*, which had most (but not all) the extensions built in. Direct support for external devices -- joysticks, and CD-audio and videodisc players -- which is provided in the Multimedia Extensions is not available in Windows 3.1. A comparison of Windows 3.1 with Windows 3.0 with Multimedia Extensions is given in Table 2.

Table 2
Multimedia Extensions Capability in Windows 3.1

Component of Windows 3.0 w/MM Extensions	Availability in Windows 3.1
HyperGuide	No
MusicBox	No
Sound Recorder	Yes
Media Player - CD Audio	No
Media Player - Videodisc	No
Media Player - Wave Audio	Yes
Media Player - MIDI Sequencer	Yes
Control Panel - Sound	Yes
Control Panel - Drivers	Yes
Control Panel - Display	No
Control Panel - MIDI Mapper	Yes
Control Panel - Screen Saver	(*)
Control Panel - Joystick	No

(*) Windows 3.1 screen savers will be different from those found in Windows 3.0 w/MM extensions.

Source: Adapted from 'NewMedia' of March, 1992

The current users of Microsoft Windows can get multimedia functionality -- including audio and video controller features -- by using upgrade kits or CD-ROM drives that include the extensions. The extensions are also available separately (without the hardware) in the form of Microsoft's Multimedia Development Kit, but this may be an expensive proposition.

For multimedia developers, Windows 3.1 provides a stable and integrated development environment. The latest version of Windows is more user-friendly and has also drastically reduced the number of Unrecoverable Application Errors (UAEs). Microsoft also promises enhanced Window stability and increased networking power.

Voice annotation feature of the MPC allows the addition of sound to word processing documents, spreadsheets, and the like. The comments that are recorded to a document can be listened to by clicking on the little microphone icon.

MPC - An 'Open' System for Multimedia Computing

By launching MPC, Microsoft has provided a much needed standard platform for the software developers to develop their PC multimedia products. To the hardware manufacturers it has provided a standard configuration for their multimedia systems. For the PC users, the MPC standard ensures that all the software and the hardware complying with this standard would be compatible. MPC is based on open standards for all its major components -- 286 or 386 CPU, VGA graphics, relatively fast CD-ROM drives, stereo sound capability, and Microsoft Windows. It is also open-ended for future technological enhancements in CD-ROMs, audio CD, MIDI, movie file formats, external laser discs, video devices, and graphics cards. The MPC standard embraces most PCs sold today and several well-known sound cards and CD-ROM peripherals.

Another point of interest about the MPC is its requirement for *Multimedia Windows* "or a binary equivalent;" this leaves the door open for future operating systems such as OS/2 2.0 to run MPC software.

MPC - A 'Compatible' Standard for Multimedia Solutions

MPC represents the first standard that is compatible with many existing hardware and software products. Multimedia PCs are available off-the-shelf from stores like CompuAdd and Tandy. Multimedia upgrades for existing PCs are affordable and provide flexible usage.

DOS is the computer platform most often designed for, as it has the greatest number of users. The goal is to allow applications that run on Windows to run on the MPC platform. Thus, MPC provides full compatibility for a substantial percentage of existing PC users and for many PC applications.

MPC Vendors - A Growing Multitude

Despite a lack of promotion, the sales of MPC titles and user tools have been gaining surprising momentum. The demand for the reference software in particular has been showing a sharply rising trend. A NewMedia report shows that titles like Software Toolworks' *World Atlas for Multimedia PC and Reference Library*, Compton's *MultiMedia Encyclopedia for Windows*, Applied Optical Media's *American Vista* and *Birds of America*, Bureau Development's *Multimedia World Fact Book* and *U.S. History on CD-ROM*, InterOptica's *Great Cities of The World* Volumes I and II, Microsoft's *Bookshelf for Windows* and EBook's *The Electronic Library of Art*, have been selling briskly. The rising popularity of these titles is expected to further boost the demand for the MPC upgrade kits and new MPC units.

CompuAdd and Tandy, the first manufacturers of the MPC computers, have already been followed by several other major manufacturers. These companies include IBM (with its Ultimedia system targeted primarily at the education and presentation market), Goldstar, Philips, Samsung, and Zenith. Several other companies are also actively pursuing their plans for entering the market for MPC upgrade kits. Workstation master Sun has been working on making its graphics and multimedia software both DOS and Windows compatible. Other companies that include Laser Computer and Leading Edge have announced plans to sell MPC compatibles. FM Towns, Fujitsu Ltd's 3-year-old multimedia platform for the Japanese market, is luring U.S. developers with its large installed base, near-MPC compatibility, and powerful graphics. Fujitsu also plans to market its systems in the U.S.A.

IBM has shipped its Multimedia Presentation Manager/2, and OS/2 2.0 version of Presentation Manager that includes multimedia extensions. Like Windows 3.1, the extensions take advantage of OS/2's 32-bit environment, including preemptive multitasking.

Microsoft's Windows 3.0 and its 3.1 upgrade, has given vendors such as Adobe, Aldus, Fractal Design and Gold Disk the opportunity to move onto the IBM platform. Meanwhile IBM-oriented DOS and Windows software vendors, facing stiff competition from products crossing over from the Mac camp, have added new enhancements to benefit the users of graphics software.

ERIC

Multimedia PC's Support of DVI and AVI (or is it vice-versa...)

A multimedia production, with its digitized sound and graphics, is a real disk hog - one second of CD-quality stereo sound takes up approximately 170K of storage. A single full-screen color image requires about 1.5 MB of storage. Without compression, a 30 frames per second (fps) full-motion video would require 45 MB per second. For the IBM PCs (and compatibles), hardware-based compression technologies like DVI (Digital Video Interactive) have provided a solution to this problem in case of video.

DVI (Digital Video Interactive) is one of the most promising hardware-based video compression technologies. DVI is a compression/decompression algorithm for full motion video and is available as an add-on board to be installed in an IBM compatible PC. A joint development of IBM and Intel Corp., DVI is both a chipset and a standard that covers everything from a software interface specification to compression. DVI is available on PCs with IBM's ActionMedia II board and now on Macintosh with New Video's EyeQ board. Ultimedia is IBM's new, complete family of multimedia solutions spanning from the fully digital ActionMedia II display and capture adapters to the high performance, MPC-compatible PS/2 M57 SLC multimedia computer. Using ActionMediaII adapters one can develop and playback applications which take full advantage of the capabilities of DVI technology.

Microsoft has tried to provide video capabilities, similar to those provided by Apple's QuickTime, through its Video for Windows (formerly known as AVI or Audio Video Interleaving). The basic configuration of Video for Windows allows the playback of video sequences in small windows (320x240 pixels) at up to 15 fps which is still not as good as full-motion video. Full-motion video is digitally recorded video played back at a broadcast standard of around 30 fps to avoid a 'jerky' image.

Apple's QuickTime provides the same functionality as Microsoft's Video for Windows, but it raises the stakes in the area of software-only video playback. By virtue of its additional compression algorithms, QuickTime allows playback at half-screen (320x240 pixels) at 15-fps or quarter-screen at 30 fps. Apple's new software also has provisions for including additional data types (besides audio and video) using a generic media handler. Due to Apple's 16-month lead over Microsoft in the digital arena, sophisticated tools that have developed around QuickTime are not yet available for Windows, but it may not remain so for long.

Multimedia PC and MIDI

MIDI (Musical Instrument Digital Interface) is a standard digital communications protocol originally developed by makers of music synthesizers. Now, it provides for synchronous control of audio, video and lighting instruments from a single computer. MIDI allows the computers to communicate with synthesizers and other electronic instruments. A significant feature of the MPC is the interface it provides to MIDI files -- one of the most accessible and interactive formats which is used for 'storing' music.

Microsoft has embedded support for significant device drivers such as the MIDI Sequencer and the Mapper as well as the Wave Audio in Windows 3.1, so non-MPC-compliant boards like the original Sound Blaster will be able to provide wavefile and MIDI support in Windows 3.1. With specifications that mesh perfectly with MPC requirements, Sound Blaster Pro is fast becoming one of the leading sound boards for MPC systems. This card, plus a CD-ROM drive and Windows with Multimedia Extensions, comprise a cost-effective MPC upgrade kit.

Object Linking and Embedding on the MPC

Window 3.1 supports object linking and embedding (OLE) which has similar capabilities as System 7's Publish and Subscribe. OLE allows intelligent integration of information from a variety of sources. OLE, which is based upon object-oriented computing, has important implications for networking and cross-platform computing. It provides for insertion or embedding of data from one document or program into another -- when the data is modified in the source program or document, it automatically gets updated in the receiving program or document. OLE fully exploits the multitasking and the inter-process communication capabilities of Windows. In particular, OLE allows multimedia objects to be embedded in the traditional Windows applications.

Market Trends for Multimedia PCs

Integrating multimedia and education is IBM's first step in its major offensive targeted at making itself the education company of the 1990s. In the words of James E. Dezell, vice president and general manager of IBM's educational systems group, "Multimedia computers will create the new knowledge of the twenty-first century codified in sounds, text, images, and graphics." IBM Vice President and President of the IBM Multimedia and Education Division, Lucie Fjeldstad, has said, "We're committed to working closely with you, the innovators of education, to offer the dynamic multimedia solutions that will equip students with the skills they will need to succeed." Recently, IBM has established its Fireworks Partners unit which will spend "tens of millions of dollars" on joint projects in multimedia technology with third parties.

Microsoft has also reiterated its commitment to MPC products that would result in -- bringing information to life and creating new and more interactive ways to educate and entertain. New developers of multimedia applications like Compton's NewMedia, and Broderbund, are showing a very positive response to the MPC standard by developing their products for the MPC platform. The authoring tools available for the MPC include well-known products such as Authorware Professional, AimTech's IconAuthor, Asymetrix's Toolbook, Owl International's Guide and Access Software's Links. Several other multimedia vendors have also unleashed a barrage of MPC titles targeted at the education and training market.

By including most (but not all) of its Multimedia Extensions in Windows 3.1, Microsoft has opened gates to a growing market of people buying PCs. Link Resources, a New York-based market research firm, estimates that 4 million PCs were sold to home users in 1991 and expects that figure to grow at a rate of 15 percent for the next two years. They also indicate that the combination of sliding prices of 386 SX-based PCs with sound cards, as well as Microsoft's decision to embed its extensions within Windows 3.1, means that there should be a sizable market for multimedia-PC applications by 1994. Link representatives predict that the market for "multimedia-capable machines" will grow from today's 5 percent of the overall PC market to 50 percent by 1996.

The emerging MPC is competing with Apple's QuickTime, IBM's Ultimedia, and several other smaller players, for a slice of the multimedia marketplace. The key players are trying to develop cross compatibilities. Apple has developed a Windows player for QuickTime videos, while Microsoft has developed a QuickTime movie converter for Video for Windows. Indeed, MPC will play a significant role in the evolution of a de facto industry standard for multimedia education.

References

Baron,D. (1992). Microsoft Offers Digital Video - Apple Betters its QuickTime, Too. *Digital Media: A Seybold Report*, 2(6), 19-20.

Bermant,C. (1992). Upgrading to MPC. *MPC World*, 1(1), 40-44.

Brown,M. (1992). Pioneering Amiga Still Packs a Wallop. *NewMedia*, 2(5), 18-21.

Buckland, M.K. (1991). Information Retrieval of More Than Text. *Journal of the American Society of Information Sciences*, 42, 586-588.

Burger,J. (1991). Making Multimedia. *Electronic Musician*, 7(11), 30-44.

Burger,J. (1993). *The Desktop Multimedia Bible*. New York: Addison Wesley.

Day,M. (1992). iOLE! *Windows Magazine*, 3(2), 79-83.

Elmore,G.C. (1991). Planning and Developing a Multimedia Learning Environment. *Technological Horizons in Education Journal*, 18(7), 83-88.

Fox,E.A. (1991). Standards and the Emergence of Digital Multimedia Systems. *Communications of the ACM*, 34(4), 26-29.

Gall,D.L. (1991). MPEG: A Video Compression Standard for Multimedia Applications. *Communications of the ACM*, 34(4), 46-58.

Hooper,L. (1993, January 21). IBM Launches 'Fireworks' Unit to Form Joint Ventures for Multimedia Projects. *The Wall Street Journal*, B6.

Jaeggli,N.D. (1992). Multimedia Systems: They Permit Students to Design Lessons from Many Sources. *The Vocational Education Journal*, 67(3), 30-32.

Keizer,G.(1992). Kids Go GaGa for Geography and History and Science and Math and.... *MPC World*, 1(1), 45-49.

Kendall,R.(1992). MIDI Goes Mainstream. *PC Magazine*, 11(6), 181-218.

Lauriston,R.,& Robinson,P.(1992). MPC Right Of the Shelf. *MPC World*, 1(1), 32-38.

Mathisen,R.W. Interactive Multimedia and Education: Specifications, Standards, and Applications. *Collegiate Microcomputer*, 9(2), 93-102.

Miller,M.J.(1992). Multimedia. *PC Magazine*, 11(6), 112-123.

Owen,B.,et. al.(1992). Multimedia PCs Here. Now. Wow! *MPC World*, 1(1), 26-31.

Pearson,L., et al. (1992). New Light On Interactive. *Presentation Products Magazine*, 6(4), 27-32.

Petzold, C.(1992). An Introduction to the Media Control Interface. *PC Magazine*, 11(1), 407-414.

Petzold, C.,et. al.(1992). Multimedia - PCs and Upgrade Kits - Do They Meet the Challenge? *PC Magazine*, 11(6), 125-177.

Polilli, S.(1992). MPC Snowballing. *Computer Pictures*, 10(1), 63-65.

Quain,J.R.(1992). Why Multimedia PCs Don't Deliver the Complete Picture. *PC Magazine*, 11(6), 132-133.

Reveaux,T.(1992). DVI Delivers. *NewMedia*, 2(8), 10-13.

Reveaux,T.(1992). Digitizers Bring Desktop Video on Board. *NewMedia* 2(8), 14-17.

Rosenzwig,S. (1992). What Exactly is MPC? *MPC World*, 1(1), 13.

Rubenking,J. (1991). Add a Musical Dimension to Your PC with MIDI. *PC Magazine*, 10(5), 355-366.

Todd,D.(1992). Windows 3.1 Saves the Day for MPC Developers. *NewMedia*, 2(3), 12-17.

Wallace, G.K. (1991). The JPEG Still Picture Compression Standard. *Communications of the ACM*, 34(4), 30-44.

Yager,T. (1992). The Multimedia PC: High-Powered Sight and Sound on Your Desk. *Byte*, 17(2), 217-226.

ERIC

NEAT: An Integrated Authoring Environment Based upon ToolBook

Stefan Mayer*
Tomasz Müldner+
Claus Unger*

Abstract

Our paper describes NEAT, an integrated authoring environment consisting of a basic ToolBook kernel extended by various OpenScript programs as well as a variety of state of the art "plugged in" external tools. NEAT supports the use of graphics, animation, audio, video, and object repositories, and provides the author with concepts such as a sophisticated, high level book, navigation, and question/answer.

1 Introduction

Sophisticated interactive courseware makes heavy use of various *media* such as text, graphics, animation, audio, video, etc., and supports a variety of powerful metaphors such as an electronic book, free/guided navigation, help-oriented interaction, etc.; see (Alessi, S. M. & Trollip, 1991, and Merrill, 1985). Authors of interactive courseware expect their authoring environments to support all these metaphors when needed, and to provide latest state of the art tools for all media used. Though there is nowadays a big variety of authoring systems available on the market, none of them sufficiently supports all metaphors, or really bears comparison with special purpose state of the art tools. Because of the big variety of media involved, it seems questionable if a single, monolithic authoring system will ever be able to combine the functions of all the corresponding single purpose tools. One might argue that most authoring systems provide scripting languages which finally allow the author to *program* a course in exactly the way he or she wants to. While this 'Turing Machine argument' does not help the non-experienced author at all, even experienced programmers are frightened by most existing scripting languages.

The arguments above led to the concept of an authoring environment which is based upon an existing authoring system (ToolBook) as kernel. Missing metaphors are realized as program modules written in the kernel's scripting language (OpenScript). Various tools can be accessed from the kernel in a consistent, transparent way. The authoring environment itself is fully embedded in the Windows environment, which in turn allows to easily exchange tools or add new tools.

In this paper, we concentrate on the added metaphors. After a brief description of the workbench approach we introduce the several concepts which have been implemented; Electronic Book metaphor, Interaction metaphor, and Guided Tours metaphor in some more detail.

* Praktische Informatik II, University of Hagen, Feithstrasse 140, D-5800, Germany
+ Jodrey School of Computer Science, Acadia University, Wolfville, BOP 1X0 Nova Scotia, Canada

ERIC

2 The Workbench Approach

In NEAT, different techniques have been used to extend the basic ToolBook functionality (see Fig. 2.1):

- Various metaphors have been added via complex OpenScript programs which, for reasons of efficiency, themselves call Pascal programs through Dynamic Link Libraries (DLLs);

- External tools (e.g. graphics designers, data bases) can be 'plugged-in' with the help of *tool adapters* which map the tool interface to the NEAT interface. Usually, each tool needs its own adapter. Whenever a tool is replaced, the adapter has at least to be adjusted accordingly. This adapter concept is well known from the HCI area, where it is used to separate the functional core from the user interface, and thus to hide changes at the functional core interface from the user interface. The functions of adapters range from simple call mechanisms to complete tool interfaces.

- Standard format objects (e.g. graphical, audio, video files) produced with external tools can be used at the NEAT interface via *use adapters* (e.g. video, audio, or animation player).

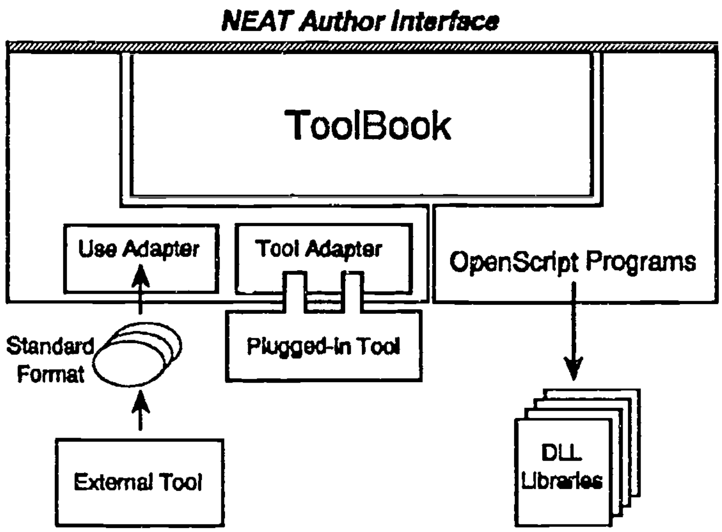


Figure 2.1. The NEAT Workbench Approach

The implementation of NEAT and the integration of external tools was supported by ToolBook's System Books concept and Windows' Dynamic Link Libraries (DLL) services. In the following, we describe the added metaphors in some more detail.

3 The Electronic Book Metaphor

In this section, we describe the hypertext tools provided by NEAT which can be used to implement courseware based on a book metaphor. The novel feature of our design is that all the tools are learner-modifiable. For example, the hyperlinks created by the author can be removed by the learner, and new links can be formed by the learner. For this reason, the user of NEAT will not be using the built-in ToolBook hotwords. Moreover, NEAT supports a distributed environment, that is on a network, each user may have his or her individual copy of hyperlinks, highlighting, etc.

Below, we describe all the hypertext tools supported by NEAT:

- ***hyperlinks***, that is electronic links between the text on one page (hotword), and the same, or different page; see also (Woodhead, 1991, and Müldner & Blondon, 1992). To create a hyperlink, the user first selects the source, and then decides whether the destination is on the same page or not. If the destination is on the same page, NEAT creates a new window that will be opened when the user activates the hyperlink. The user (the author or the learner) can type a text in this window. Thus, this type of a hyperlink implements *help windows* and *footnotes* (see Figures 3.1 and 3.2 below). If the user decides on a different page, the activation of the hyperlink will move the user to that page. Thus, this type of a hyperlink implements a *forward* or *backward* reference. Since page numbers give a limited information (the same page may have different page numbers in two different guides), the destination page is selected from the table of contents. To help the learner to recognize a hyperlink, its source is underlined in a different color depending on its type, for example in red for a forward reference. Hyperlinks are created in the same way by both the author and the learner. To help the user to create hyperlinks, he or she may create *unfinished* links by inserting the source to the list of sources. Later, the user may decide on the final shape of the link by selecting the source from the source list and linking it with the current page.

  Formation of links between pages may create conflicts between this structure and the structure of guides described in the previous section. For example, the author, or the learner may create a hyperlink between two pages that do not belong to the same guide. This problem is solved by not allowing the user to create such a hyperlink.
- ***highlighting*** of a part of text on a page. The user (again, both the author and the learner) can select the desired color from the palette of colors, and select text on a page, in order to highlight this text; that is to change its background color to the selected color. Highlighting can be added and removed at any time. A page occurring in more than one guide may have a different highlighting in each of these guides.
- ***user-modifiable index*** resembles a standard index in that it can be used to find a reference to a phrase in courseware. In addition, this index can be modified by the learner, who can remove the existing phrases or references and add new ones. Unlike the traditional index, NEAT index does not show the list of page numbers associated with a reference. Instead, if the user selects a reference, she or he will be moved to the first page with that reference and will be able to traverse the list of all such pages (see Figure 3.3). This list is modified after each traversal, and most recently visited pages are stored in front of the list so that in the future traversals the user will have a quicker access to these pages.
- ***notes***, include both ***margin notes***, which may have different contents on each page, and ***global notes*** which have the same contents for the entire courseware. The learner can activate one of these *notes*, for example using the specialized icon, and then read the text included in the notes, or modify this text. Thus, notes resemble annotations, and they can be used by a single learner, a group of learners, or even as a communication medium between the instructor and the learners. In a distributed environment, it is possible to send notes through the electronic mail to another user.
- ***history*** mechanism that allows the learner to move to any page that has been visited. The history is shown as a list of names of visited pages.

- ***bookmarks***, which are hidden, labeled markers that can be placed anywhere in courseware. The user can access the list of names of existing bookmarks, and then move to the place associated with a desired bookmark simply by double-clicking the name of this bookmark. Clearly, bookmarks are associated with guides, that is for each guide there may be a different set of bookmarks.
- ***table of contents*** with ***bread crumbs***. During authoring, the chapter and section titles can be designated, and they will be inserted by NEAT into the table of contents. This table will have a hierarchical structure, showing the list of chapters, and unfolding sections belonging to a specific chapter when the learner clicks on that chapter. The user can select any page and move to it. For each guide, there is a different table of contents. NEAT will also trace the user activities and place bread crumbs next to the chapters that have been visited. Using hyperlinks, the user may visit only parts of sections and so a bread crumb is in the form of a black rectangle which indicates what percentage of the section has been visited.

Below, we provide several snapshots of pages created with NEAT. Assume, that the user wants to create a hotword for the words "Principal file" in the text shown in Figure 3.1. First, the user selects this text, and then invokes a specialized *Hotword* icon. This icon gives the user a choice of adding a link to a page, or to a window, storing "unfinished" link (described later), or removing hotwords. Assuming that the user has chosen to open a window, the screen will look as in Figure 3.2.

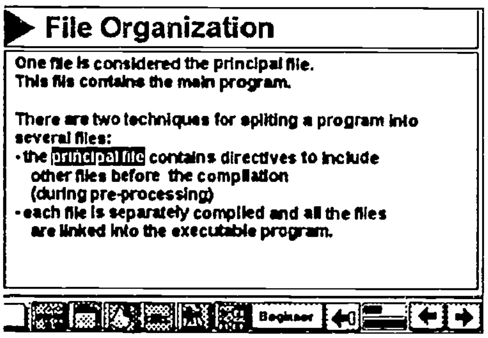


Figure 3.1. Selection of a sample text

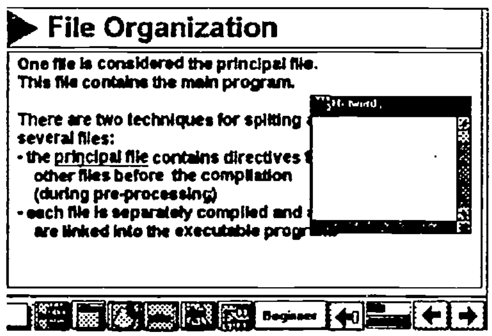


Figure 3.2. Hotword window

The user can type any text in this window. After the hotword window is closed, the hotword remains underlined, and the user can click on it any time to show the hotword window. In Figure 3.3 we show a traversal of the list of pages associated with the word "C". Figure 3.4 shows the table of contents.

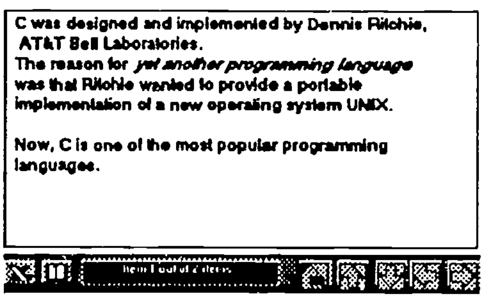


Figure 3.3. Traversing the index

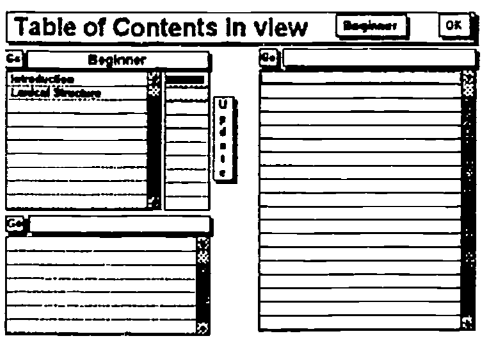


Figure 3.4. Table of contents

4 Interaction with the Learner

Questions form an essential part of any courseware and are important means to interact with the learner. NEAT supports interaction concepts missing in basic ToolBook. It provides templates for several kinds of questions, and supports various ways of answers and answer-dependent help without forcing the user to write ToolBook scripts.

The author can choose from the following templates:

1. **Multiple choice** template, which provides slots for a number of *choices* with check boxes associated with each choice.
2. **Fill-in-blanks** template, where the learner has to fill in missing pieces of text.
3. **Numeric** template, where the answer has to be in a certain numeric range.
4. **Position analysis** template, where the answer is provided by pointing, or dragging an object, to a certain part of the screen.

In the following, we concentrate on multiple choice and fill-in-blanks interactions.

In case of *multiple choice interaction*, there are two general ways an author may deal with learners' answers. He can associate certain *combinations* of choices with individual replies (*context-sensitive* reply), and/or associate *single* choices (or non-choices) with their corresponding replies (*context-free* reply). If for a certain answer, i.e. combination of choices, the author has not provided a context-dependent reply, all corresponding context-free replies are given to the learner.

In case of *fill-in-blanks interaction*, the author can define sets of answers by using global interaction attributes (e.g. relaxed case, spelling tolerance, allow unnecessary words) or by using a simple answer grammar which allows to specify wild cards, embedded strings, synonyms, optional words, etc. Each set of answers can be associated with a detailed multi-media reply.

In the following, we show how the author can interactively define a multiple choice interaction:

1. Choose *Create Question* from the *Question* menu and select *Multiple Choice Question*.
2. Define the layout of the ToolBook question page.
3. For all context-sensitive replies:
   3.1 Define a combination of check boxes
   3.2 Provide an individual reply which may use text, graphics, audio, video, database access, etc.
   3.3 Specify a score for this answer.
4. Define the layout for a context-free reply page.
5. Associate textual, graphical, etc. objects of the reply page with single choices/non-context-sensitive reply choices.

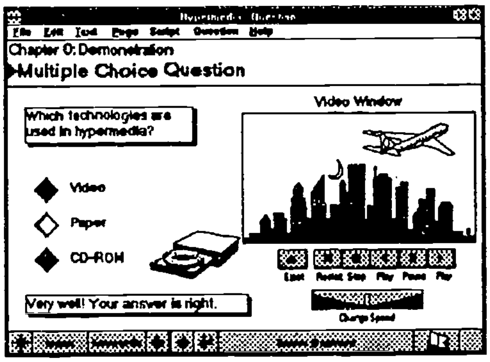


Figure 4.1. Multiple choice

6. Define the Correct Answer reply and Hint reply (as in step 3.2).
7. Test the interaction
8. Install the interaction.

Figure 4.1 shows a context-sensitive reply on a multiple choice question, and Figure 4.2 shows the layout of a multiple choice question page containing all possible context-free replies (which in reality spatially overlap). Figure 4.3 shows the individual context-free reply on a multiple choice question, and Figure 4.4 illustrates the use of the answer grammar.

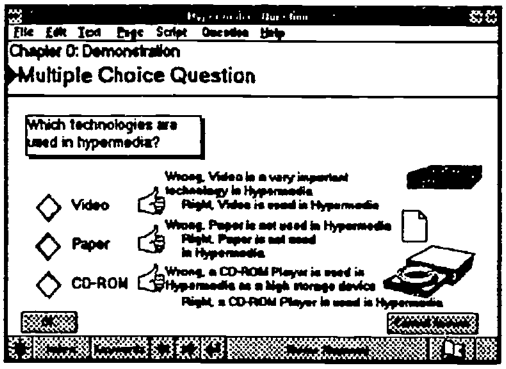


Figure 4.2. Multiple choice question page with context-free replies

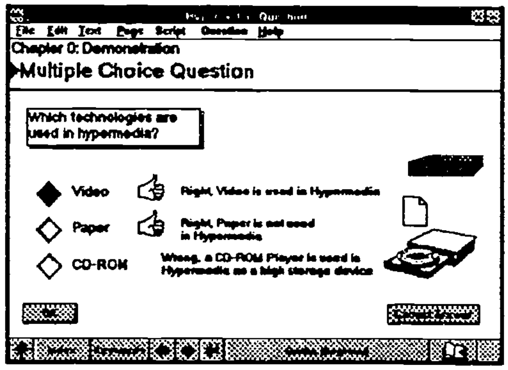


Figure 4.3. Multiple choice context- free reply

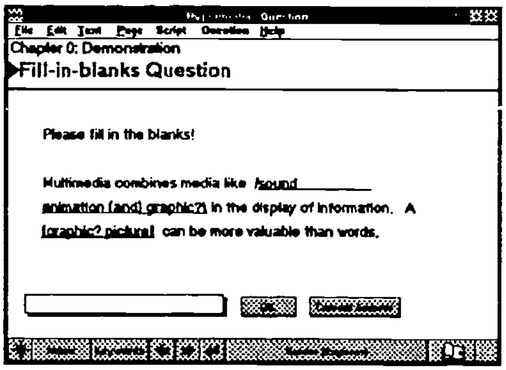


Figure 4.4. Fill-in-blanks question page

5 Guided Tours

Like almost all existing authoring systems, ToolBook is lacking in flexibility to link pages. Without extensive programming in OpenScript, it is impossible to define for a single course a variety of guided tours for different user classes, for example beginners, experts, etc., and let the learner dynamically switch between these tours.

NEAT provides an author-friendly two level mechanism for defining guided tours:

1. Each *page* has a local *guide attribute* specifying to which guided tours this page may belong. The guide attribute is set by the author when developing the page.

2. The author may group *consecutive* pages into *page groups* and define guided tours in terms of sequences of page groups.

A guided tour is defined by its constituting page groups. If a page, according to its guide attribute, belongs to a guided tour that does not include a page group containing this page, the page will never be visited during this guided tour. When a page group is entered during a guided tour, only pages with the appropriate guide attribute are visited, while all other pages are skipped.

The following two figures illustrate how an author can interactively define guided tours. In Figure 5.1, the left list box contains a list of pages and already defined page groups. The right list box contains all pages of the most recently defined page group together with their guide attributes. Whenever the author clicks a page group in the left box, its pages are shown in the right box. Figure 5.2 illustrates how page groups can be linked. The upper left list box shows all existing page groups. Also, all available guides are shown. The author can perform the following actions:

- select a guide
- select page groups (nodes) from the list box
- arrange page groups on the screen
- for the selected guide, define links between page groups.

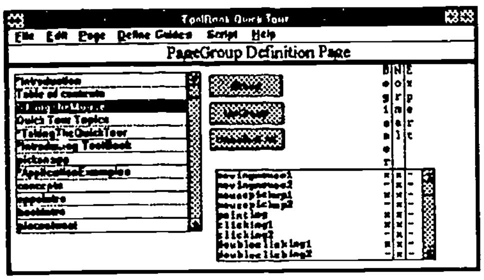


Figure 5.1. Defining page groups

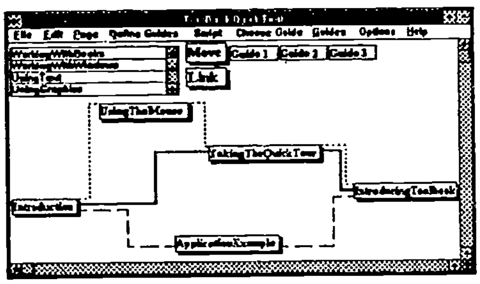


Figure 5.2. Linking page groups

The author stepwise builds up a graph containing all guided tours. In the current version, guided tours are restricted to a single ToolBook book. In future versions, NEAT will also allow to link page groups from different books. Although the guided tour approach is defined for guided navigation, it will also be used to support free navigation and to provide the learner with information helpful to select the next page to be visited.

Conclusion

We described the NEAT workbench approach with special emphasis on the high-level electronic book, interaction, and guided tour metaphors. All components have been implemented. Our early experiences with NEAT are very encouraging. Using NEAT saves a lot of time when producing courseware and significantly increases the quality of the courseware.

Acknowledgments

This paper was written while the second author stayed at Open University in Hagen, Germany, partly supported by the grant from Deutscher Akademischer Austauschdienst. This author would like to express his appreciation for the hospitality of the Praktische Informatik II Department, University of Hagen; in particular to Prof. C. Unger and Mr. S. Mayer.

References

Alessi, S. M. & Trollip, S. R. *Computer Based Instruction* (second edition). Englewood Cliffs: Prentice-Hall, 1991.

Merrill, M. D. Where is the authoring in authoring systems. *Journal of Computer-Based Instruction*. Autumn 1985, Vol.12, No. 4, pp. 90-96.

Müldner, T. & Blondon, R. Design and Implementation of Courseware for Teaching Programming Languages. *Proceedings of the fourth International Conference on Computers and Learning*, Wolfville, Canada. Berlin: Springer-Verlag, 1992.

Woodhead, N. *Hypertext and Hypermedia*. Wilmslow: Sigma Press, 1991.

An Instructional Design Advisor for Computer-based Learning: ISAAC

RAY MCALEESE AND LAI-MUI CHING [1]

Institute for Computer Based Learning, Heriot-Watt University,
Edinburgh EH14 AS,

SUMMARY

The paper reports on an attempt to build a computer-based learning advisor to assist authors designing computer-based courseware.

The Instructional Strategies-Analysis Advise Companion [ISAAC], comes from a needs assessment to have an adequate environment for the instructional design in computer-based authoring. Presently, there are a number of tools for the computers that enable computer-based authoring work to be carried out [- authoring languages e.g. ToolBook, Authorware, IconAuthor]. These tools range from simple, generic text/graphics tools to complex, multimedia software applications. There are also auxiliary tools or advisory systems that help designers to choose appropriate instructional strategies [e.g. IDExpert]. However, these two types of tools are normally *not integrated* together, though they both may be used in the instructional design process. The disadvantage of having them as separate systems/tools lies in that the advisor is capable of only giving *passive* advice, but not capable of helping the users *during* the actual process of instruction design. The idea behind this work is to have an instructional design system that has both environment/tools for the authoring of instructions as well as an advisor which can render advice voluntarily to user when the user's instructional plan deviates from that of a chosen instructional theory. An issue being addressed is that of advisor-intervention in the design process. The approach reported in this paper has been to develop a prototype based on a scaled-down domain knowledge using one instructional design theory [Gagné].

Our work summarises advice giving within the context of CBL authoring arguing that by and large, the trend in advice giving is towards providing 'usable' advice. That is, having advice which is meaningful in the context of the user's task, while at the same time, it is given at appropriate time, in the case of active advice or a help system. Such an advisor requires a more demanding capability, such as the capability to track user's plans, modify advice based on user's feedback, and reason about user's actions and apply and understand natural language in communications with the users.

The paper concludes that:

- The idea of having a computer-based authoring system running with an on-line active advisor has been demonstrated in ISAAC;
- In spite of being a prototype system, ISAAC carries a vision about future authoring tools. ISAAC's structure and design can be extended to become a complete, usable and powerful adjunct to authoring systems.

---

1 Lai-Mui Ching completed the work on ISAAC as an MSc student in Heriot-Watt University, in 1992. The work was supported by a number of colleagues in ICBL - Terry Mayes, Roger Rist, Patrick McAndrew and Kate Tuckey.

ERIC

INTRODUCTION

There are numerous tools available that facilitate instructional design. Within the context of a computer-based system, these tools range from simple generic text-based or graphics-based to complex, multimedia or hypermedia-based. For example, anyone who wants to plan a lesson may use a general-purpose word-processor, like the Microsoft WORD™, to write his/her plan. Alternatively, he/she can use a graphics tool, like the MacDraw™, Inspiration™, or XFig™ program, to plot his/her lesson plan graphically. Either way, the designer will design his/her lesson to the best of his/her knowledge and capability. The resultant effort - the lesson plan, need not necessarily be the optimal or considered to be sound pedagogically. It depends very much on the designer's experience and knowledge of pedagogy principles and theories.

At the other extreme, where the designer has access to highly complex and rich instructional design tools or systems, like the prototype IDE [Instruction Design Environment] [2][IDE Reference Manual, 1988] and Magpie™[3], the designer is given a wider range of tools and more specialised environments to work in. Such systems usually have the advantage of having better information management capabilities, which help the designer to keep track of his/her plans, subject content organisation and references. Such capabilities are important and essential. Take IDE for example, it is a workbench for instructional analysts, designers and developers. A number of tools are provided by the IDE for the purpose of instructional design. Tasks such as data collection, task analysis, knowledge structuring, storyboarding, and script layout are supported by the system. Such capabilities will definitely enhance the instructional design process. Such useful capabilities can only assist the designer as far as information management task is concerned. They do not, contribute to the pedagogy of the instructions produced.

There are other side effects of having complex, powerful tools. One side effect is that such tools can become hindrances to the main focus of the designer. The "value-added" capabilities may sometimes serve to distract the designer and divert his/her attention from the actual task of planning a sound lesson. For example, the designer may end up spending more time on learning to use the tools rather than concentrating on his/her design work. He/she may also spend more time trying out all the "bells and whistles" provided by the system.

Beside these authoring tools, there are systems like the existing advisory systems, which provide static advice for the instructional design. They do not assist the designer directly in the process. They are generally stand-alone systems. Many of them would be built on top of expert systems [the latter which are normally simple rule-based systems]. Such systems typically pose series of questions to the designer in a Socratic style. Depending on the designer's answers, different advice would be given to the designer. One example of such advisory system is the HyperTactics [Jonassen and Harris, 1991]. This kind of advisory system serves to help the designer in the process of instructional design. Designers can still get effective advice and recommendations through consulting these systems, since the pedagogical knowledge on which they are based has been elicited from an expert instructional designer.

The only drawback, or rather inadequacy of this type of system lies in the very fact that they are **independent** of the actual process of instructional design. The advisors are only capable of rendering general advice. They do not include considerations of what the designer's actions are, his/her situational goals, errors and deviations. The advisor can only assume that the designer will take the advice and know how to apply it correctly. Moreover, effective application of strategies in instructional design only occurs if the designer has done a proper instructional analysis of the subject domain. The systems will not be able to correct or guide the

---

2 IDE was a prototype design environment built at Xerox·PARC by a group that included Dan Russell and Dan Jordan.

3 Magpie™ is a multimedia authoring software for computer-based courseware development. It is built for the UK Acorn computer.

designer on instructional analysis. Given that such systems only give advice based on what the designer has answered in the series of questions it will be difficult to visualise how a novice instructional designer can learn by going through the consultations. He/she cannot try out the design and get the advisors to comment on it. The latter way of obtaining advice is a typical example of how normal means of seeking/giving advice is done. Experts and novices often work in this way in a non-computer environment. This active advisory aspect is absent in consultation systems.

There are many good reasons to justify the employment of an authoring system, and an advisory systems together. At present, in the training industry, there are strong demands for fast production of training/courseware materials. The deployment of computers in industry has also led to many training materials being delivered on computers. In order to meet the demand for fast production of courseware, which at the same time, is pedagogically sound lies in providing courseware designers with support in the aspect of writing competent and pedagogically sound instruction. Many instruction designers will be experts in one or more specialised field, that is, the subject domain, but very rarely are they experts in the area of instructional design. Providing support will mean making available consultants or advisors from whom sound recommendations and guidance can be rendered when necessary. Having a human advisor [one who is an expert in instructional design] is ideal but such experts are rare. Therefore, to compensate for this lack of experts in instructional design, computer-based authoring tool should take on the role of supporting instructional design at the pedagogy level.

INSTRUCTIONAL DESIGN

Any advisory system must take an instructional design stance. Designing effective instructions for the purpose of supporting the learning of an individual student is the ultimate aim of instructional design. Many theories, practices and models been developed through the past two decades. These may be considered formal approaches towards instructional design, although few of them have not undergone any real testing and evaluation. Many existing models are the result of adaptations made to other models to suit a particular learning environment. The general tasks involved in many models are generic. For example, tasks like the setting of instructional objectives and task analysis may be phrased differently in different models but they are generally recommended steps to take. At the same time, it must be noted that it is not necessary that learning theories are explicitly prescribed in many of these instructional design models. We take as our starting point the Gagné and Briggs Model.

The Gagné and Briggs Model

The instructional design model that is used in ISAAC is the model which is designed by Gagné and Briggs [Gagné et al, 1992; Briggs, 1977]. The model is based mainly on the systematic approach for designing instructions [Dick & Carey, 1990] and Gagné's conditions of learning theory [Gagné, 1985]. The systematic approach to instructional design suggests that instructional design should be conducted by *means of a systems approach*. This approach basically includes a series of steps for the instruction design process and a feedback mechanism at each step of the design for validation purposes. The steps include the analysis of needs and goals, followed by the analysis of the instructions and entry behaviour. It ends with an evaluated system of instruction that will test for success in achieving set goals.

In ISAAC, the systematic approach is simplified to emphasise instruction [adapted from Dick & Carey, 1990]. the setting of instructional goals, the instructional analysis and the instructional strategies steps. It has been suggested that the best way to design instruction is to begin with the identification of the human capabilities to be established for training. Human capabilities can be broadly classified into the five learned capabilities. Each of these capabilities represents a type of capability to be learnt. Each of the type of capability, in turn, has specific requirements that have to be met to ensure its effective learning. These requirements are somewhat independent from the actual content of the subject matter [Gagné, 1991]. Specific rules (to be built into an advisor) can be applied to each category which will identify those

ERIC

external conditions that will best support the internal processes of learning of the respective category of capability.

In fact, Hannum has interpreted these rules as structures [Hannum, 1988]. For example, for the domain of motor skills, the structure is a linear sequence. For the domain of intellectual skills, the structure is a network [similar to semantic nets]. For the domain of information, the structure is likely to be a hierarchical or tree structure. However, according to him, there is no apparent structure for attitude or cognitive strategies. The latter, in fact, is considered to be an internal condition of the learner [Gagné et al, 1992]. In this way, these identified types of learning outcome can be used to classify instructional objectives as well as to derive the instructional plan [which is referred to by Gagné as the *instructional strategies*] for supporting the learning. Since, it is often that a single course will contain more than one objectives, the ability to classify these objective. actually simplifies the instructional design process. The Hannum structures are appropriate to a structured advisory system. Before we describe in detail, ISAAC, we need to consider the general principles of advice giving.

ADVICE GIVING

ISAAC attempts to support the process of advice giving. An advisory system requires **advisory knowledge**, a term used by Carroll [Carroll, 1987; Carroll & McKendree, 1987]. They noted that within this advisory knowledge, there are three kinds of knowledge involved:

- general skills
- domain knowledge
- a user model

Carroll and McKendree described the first type of knowledge as the skills required in advice-giving. It includes strategies for giving advice [advice strategies] and communication capability. By advisory strategy, it refers to the tactics of giving advice. The second kind of knowledge, the domain knowledge, is the task-related knowledge. That is, it is the knowledge or information which the user is advised on. It forms the content of the advice. The third kind of knowledge, the user model, is the knowledge about the user and the context and activities which the user is engaged with. Taking Carroll's point of view, the design of a usable advice, ideally, has to include all these three kinds of knowledge.

Different Approaches Towards Advice-Giving

There are different approaches proposed by various workers in this area. The approach which is supported in this paper is that of Carroll & Aaronson and also that of Fischer.

General Skills

General skills refer to the strategies of advice-giving and the communication capability. A typical approach towards advisory strategies is to adopt a single advisory style that offers little or no empirical rationale. One such approach is the Socratic style of question-answer; another approach adopts the learning-by-doing environment. In the case of the Socratic style, the system poses questions and the user is expected to answer or vice versa. This approach will result in a system-driven [system-controlled] dialogue where the user is directed and passive throughout the session. Giving the system the control of the dialogue may actually *undermine* the usability of the system. Moreover system-controlled advisors rarely include *explanations* or checks that the user really understands or applies the advice.

Learning-by-doing environment, on the other hand, is quite different from that of the Socratic style. In this approach, the dialogue is user-driven or user-controlled. The user initiates actions and the system compares the user's actions to that of an expert. It then formulates feedback based on the comparison. Having such capability to design feedback according to user's actions will imply more demanding requirements than that of the Socratic style. The system must know all the implications of a user's action with respect to an expert action, as well as the significance of any potential departure from the expert's action. However, this approach does not typically capture general knowledge about strategies for error correction. Both the Socratic style and the learning-by-doing approach lack any real empirical testing and validation.

Clearly, in these two approaches, the strategies are not explicit from the domain knowledge. They merely form part of the system code. In contrast t › these two approaches, Sleeman and Brown [1982] along with many others in the ITS field indicated the importance of making the advisory strategies explicit. Parallel to their ideas is Clancy's claim for the separation of the domain knowledge from the advisory strategies. Fischer [Fischer et al, 1985] advocated yet another approach for advice-giving. They recognised the need for *a variety* of advisory strategies. Advice strategies, according to them, can be classified into two broad categories, *the passive strategy* and *the active strategy*. Passive strategy means that a system only gives advice when the user requests it; that is, it is user-initiated. Active strategy, on the other hand, is system-initiated; the system, at certain points of user's actions, renders advice to the user without the user's requesting for it. This intervention by a system is a crucial aspect of advisory systems. Here the locus of control is with the system.

In most systems, the use of passive advice is more frequently used than active advice. One obvious reason is that the active strategy demands the system to be capable of discerning the implications of each and every user's action in order to know when to give the advice. There are two implications of this capability: first, it means that the system needs to recognise the user's intent and plan of action in order to provide appropriate advice. The inference of intent is still [and will probably remain] a "black art". Secondly, since the usual time for giving advice is when an error or a potential error is committed the user, the system needs to know what actions constitute an error or a potential error. The system needs to reason about the user's behaviour and also his/her goals [Fischer et al, 1985]. Knowing user's intent and ability to reason about it are both difficult and controversial issues. Carroll and Aaronson observe in their experiment on SmartHelp [Carroll et al, 1988], that the user's goals are in "constant fluctuation". It is difficult therefore for the system to "guess" what the user intends to do

Domain Knowledge

In most instances of any advise or help facilities, there is a need to include domain knowledge. The difference among various approaches normally lies in the different levels of emphasis which are placed on the design of the content on the advice. In more recent approaches, there is a trend for context-sensitive advice and help alike. This implies that the advice will have to designed to incorporate, not only the domain knowledge appropriate to the context but also the right amount of knowledge. This makes ISAAC's design more complicated and demanding. Carroll et al pointed out several issues relating to the domain knowledge. These issues include the psychological pertinence issue, the issue of multiple representations of advice and the grain of analysis. The psychological pertinence issue outlines the need for the system to represent its knowledge as *people know it* [Fitter, 1979]. This issue actually calls for the interpretation of the user's actions with regard to his/her intention. This area of concern involves other aspects such as the human cognition and the computational knowledge techniques.

The idea behind multiple representations is that it seems it is necessary to represent any domain in multiple ways [Fikes, 1981; Stevens et al, 1982; Spiro et al, 1991] so as to provide enough flexibility to support effective advising. Multiple representations could include rote description, functional description and procedure teleology description. Another way of looking at multiple representation is the need to consider different levels of knowledge to be presented in the advice [Lloyds, 1990]. According to Lloyds, there are in general two levels: the near transfer knowledge and the far transfer knowledge. The availability of any of the two levels of knowledge to the user, is dependent on individual user differences.

The grain of analysis, according to Carroll et al, is an issue referring to the issue of *how little domain knowledge, and at how large a grain,* is optimal. There have been different views on the grain size of the domain knowledge to be represented. The traditional view is to incorporate as much knowledge as possible in an advice; the current trend, however, emphases representing scenarios to classify user situations and setting the grain of the advice at that level [Carroll and McKendree, 1987].

User Models

There is a growing demand for context-sensitive type of advisory systems. Context-sensitive advice can also be considered as individualised advice. To provide for such advice, ISAAC needs to monitor and analyse user activities. There have been few approaches developed. Examples of these approaches include the use of the normative user model , vocabulary analysis , behaviour analysis and individual differences. One example of the normative user model is the training wheels interface [Carroll and Carrither, 1984]], which makes use of an assumption that new users of a word-processing application do not require specific advanced system functions. This assumption is based on an evidence that new users often access advanced system functions by mistake and they often ended up confused and distracted. Vocabulary analysis is used to deduce the user's skill level [whether he/she is a novice or expert] through his/her vocabulary usage [Rich, 1983]. Here, keywords are analysed. Yet another way to deduce the user's skill level is to examine the user's behaviour during the entire task. This would require the monitoring and evaluating the user errors. In one instance, this is carried out by using stored prototypical error patterns as the user models [Coombs and Alty, 1984]. This latter approach is more complicated and it is also, at the same time, being implicated by an established area of work, bug analysis theory. The issue of individual differences is a difficult one. Although catering of advice according to individual differences is a consideration of undoubtedly immense importance for advice-giving system, it is presently still an undeveloped area [Carroll et al, 1991].

Other Approaches Towards Advice-giving

Another type of approach is one that tends to place more emphasis on the presentation levels of the advice and help. The approaches mentioned so far have not included any direct emphasis on issues like screen layouts and presentation. This particular of approach generally provides more insight and considerations to the aspect of the presentation. There is a tendency to classify this approach as an approach for the interface design. However, they can still be considered to belong to the advice-giving domain as they are very important factors for the usability of the advice. One example of this approach can be seen in the implementation of an on screen help for a library data base system [Saule]. Beside some brief references to the knowledge considerations, a lot of emphasis have been placed on the presentation level. This approach outlines several considerations which otherwise may be ignored. As compared to other approaches, this approach may seem more trivial in the aspects it emphasises. However, it does address some aspects of advice [and help] which are contributing factors to the usability of ISAAC.

What Do We Really Know About Advice-Giving?

In most existing advisory system only a single style is adopted with no explicit usage of advisory strategy. Even with the strategies proposed, as in the examples outlined these strategies are seriously lack empirical validation. By and large, the trend in advice giving is towards providing '*usable*' advice. Usable advice refers to having advice which is meaningful in the context of the user's task, while it is given at appropriate time. Such an advice will require more demanding capability, such as the capability to track user's plan, modify advice based on user's feedback, reason about user's actions and apply and understand natural language in the communication with the users. There are many considerations to be taken in the design of an advisory system. For ISAAC, only a few of the strategies outlined here are actually implemented. One of them involves the provision of the procedural and conceptual levels as proposed by Lloyds. The provision of both active and passive advice, as suggested by Fischer, is implemented in ISAAC, although some of the strategies suggested for the active advice, like the min-message-delay and the max-message time, are not used in ISAAC. Only the immediate feedback strategy is used. The last strategy used in ISAAC is the choice of language and terminologies used in the advice message itself. This strategy governs the way the advice is being phrased and it ensures that unnecessary use of instructional design or computer jargons are avoided.

INSTRUCTIONAL ADVISORS

In this section we review briefly some of the work that has been carried out with instructional advisors. [4]

IDE

The Instructional Design Environment [IDE] [IDE reference manual, 1988] is a workbench where a collection of tools is provided for the instructional design task. These tools aim to simplify the job of an instructional designers by assisting him/her in the different aspects of the instructional design task. The different aspects include data collection, task analysis, knowledge structuring, storyboarding and script layout. Basically, IDE assists the instructional designer in the information management aspect of the design process. It is built on NoteCards. Collaborative group work is supported by this system, making possible the co-ordination of instructional design in a team. The functionalities of IDE include the capability to manage large amounts of data, the provision of representations of task and domain analysis, the ability to organise the content of a course and the capability to sequence the instructional material for delivery. The goal of IDE is to minimise the time and resources required to develop instruction and improve the consistency and quality of the instruction. IDE may possess the necessary tools for information management and may also reduce the time to develop the instructions, but it does not provide the tools for the ensuring of the quality of the instruction. One reason is obvious; IDE is neither theory based nor model based.

ID Expert

ID Expert or Instructional Design Expert [Wilson, 1990/91] has been developed for the Army Research Institute by Merrill and Zhongmin Li of Utah State University and Human Technology, Inc [e.g. Jones et al, 1990]. It is implemented using Hypercard™ from Apple and Nexpert™ from Neuron Data for the Macintosh™. ID Expert is an expert system that is capable of providing support specifically on the subject matter and authoring environment. It is also capable of reasoning about the information it collects from the user using the rules stored in a knowledge base. It begins by collecting information about the subject content, the goal and attributes and compiles these information into a coherent instructional strategy. Unlike IDE, ID Expert is model-based. It is based on the elaboration theory proposed by Reigeluth and Stein and component design theory for developing micro-level strategies by Merrill. In this way, the system is capable of providing the designers more specific and concrete support for the planning of instructional strategies based on the learning outcome. However, ID Expert is content specific and it operates in a batch-mode. ID Expert is capable of recommending advice on strategies *only after* all the information are given to it. *It is not an active advisor.*

IDD Advisor

The IDD Advisor or Instructional Design and Development Advisor [Wilson, 1990/91] is designed to provide support for intelligent job aid for instructional designer, teacher, or course writer in the process of instructional design. It also provides an intelligent front-end to a data base that contains the components of the instructional sequence being developed. IDD Advisor makes use of sets of rules in a rule-based expert system to guide the user in the instructional design and development process. The rule sets include those rules that will assist in specific tasks like performance analysis and needs analysis. These tasks are further divided into sub tasks such as task analysis, objective writing, write test items, instructional design and media selection. At each of these sub-tasks or tasks, the advisor will help the user to identify the required information or techniques. For example, in the task analysis sub-task, the user will be assisted in the selection of the most appropriate task analysis technique.

---

4 See also Hobbs, 1990; and Korionos, 1991.

ERIC

This system has, to some extent, some basis of theory in it. For example, the task analysis techniques that the advisor will recommend to the user are based on that proposed by Jonassen, Hannum and Tessmer [Jonassen et al, in press]. The entire design and development process follows a certain order which, in some ways, is pretty similar to that of the systematic approach. The entire system has been developed for the personal computers [IBM or compatibles]. The system, as a whole, seems to be a good tool for instructional design. However, there does not seem to be any indication that this system can become a learning tool for a novice designer; that is, it is doubtful as to whether this tool will allow novice designers to gain some insights and understanding to the actual design process, especially the instructional analysis and instructional strategies. As the system's knowledge base is built using a rule-based expert system shell, VP Expert, the kind of advice and explanation provided by the system will most probably be in the form of "rule dumping". That is, the design of the advice and explanation will be limited to the capability of the expert system shell it employs.

ODE

ODE is one of the NESTOR[5] tools, which were developed for the purpose of computer-based authoring. It is a structure editor which is designed for instructional analysis. ODE has been designed to function together with the other NESTOR tools. In the functional specifications of ODE it will be linked to a central library where all the information about strategies will be found. At the same time, ODE will be integrated with the resource management module that will enable the selection of media and resources. It is important to note that ODE has an implicit design model embedded. This is the system designed by Horne and called Information Mapping. As ODE is developing no further details will be provided here.

HyperTactics

HyperTactics [Jonassen] is a built on HyperCard™ and Level 5™ expert system shell. It is an advisory system that provides advice for the planning of instructional strategies. HyperTactics functions mainly by presenting a series of questions to the user. On the basis of on the answers collected from the user, the system reasons using this information and infers appropriate strategies for the user. At every step, it provides explanations i the form of "canned text" that explains the definitions of terminologies and context from a perspective of the instructional theory. HyperTactics has been designed to be a stand-alone advisory system. It is not integrated with any authoring tool. It only provides advice based on the information [answers] which the user gives to the system. It is clearly model-based; it begins with the aim of trying to establish the type of learning outcome [hence, performance objectives] for the course and it ends by asking about the type of media or presentation the user needs to use. It follows the systems approach. Being a stand-alone system it has some disadvantages. It is not capable of assisting the user in the actual process of the design and development. It is not context-sensitive. It can only "trust" that the user will understand its recommendations and apply them correctly. It does not assist the user beyond providing the recommendations on strategies. The user will have to work his/her instructional design using some other tools. However, HyperTactics is trying to provide sound advice to the users without taking the control from the user. It can be contrasted to ID Expert, at the other extreme, which takes the entire design control out from the user's hand.

CONCEPTUAL DESIGN OF AN ADVISOR

Information [and Knowledge] Modelling

One part of the conceptual design in ISAAC is to study and analyse the gathered information to gain a deeper insight and understanding into the process of design. This process involves two types of information : [1] information gathered from the task analysis, [2] knowledge extracted from Gagné-Briggs instructional design model. The latter form of information is needed for the development of the knowledge source which is required by the system in the advice-giving process. In the design of this prototype, only a subset of the instructional theory

5 See the paper by Skyes et al in these proceedings for details of ISLE and the Nestor tool set - ISLE: a collaborative project to build an Intensely Supportive Learning Environment

will be used. The information modelling is done by taking each high-level task and analysing the information involved. There are two high-level tasks in instructional design, which are:

- the instructional analysis task, and
- the task of applying instructional strategies.

Instructional Analysis

In general, the typical input to an instructional analysis process [IA] contains the following information:

- training requirement
- subject matter
- a student model.

Usually, during the course of deciding the instructional objective, the designer will also plan for the training requirement. The latter typically consists of data like the level of training required, entry prerequisite, length of training needed and so on. Such information are important and it must be noted that all of them do not necessary involves planning; they could be fixed beforehand. Nevertheless, all these information will contribute to the designer's task of writing the instructional objective, but one of the most vital part of this activity of writing the instructional objective is the deciding for the type of learning outcome of the course. The type of a learning outcome may be classified by one of the five types of learning domains defined by Gagné [Gagné, 1985]. The subject matter, on the other hand, constitutes the instructional content. The components of the instructional content are identifiable by specific types which relate to specific learning outcomes . The types can be a rule, concept, part-skill or fact. One of the five learning outcomes is actually itself an intellectual skill. There is a special relationship between the "intellectual skills" and the "cognitive strategies". The cognitive strategy is a special kind of intellectual skill, with particular importance to learning and thinking [Gagné et al, 1992].

The instructional analysis will result in an organisation of the instructional content into certain structures [which has been termed knowledge structures], depending on the type of learning outcome. These structures have been broadly identified to be :

- linear, sequence structure
- tree structure
- graph or network structure [Hannum, 1988; Venezky et al, 1991]

The components of the instructional content relate to each other in certain ways. According to Gagné's theory [Gagné, 1985], the way which the components relate to each other depends on the type of learning outcome defined in the instructional objective. Information relating to the student model can also be attached to the instructional objective. However, the student model will require more complex analysis and representation than information, like the training requirements.

CONCEPTUAL DESIGN OF ISAAC

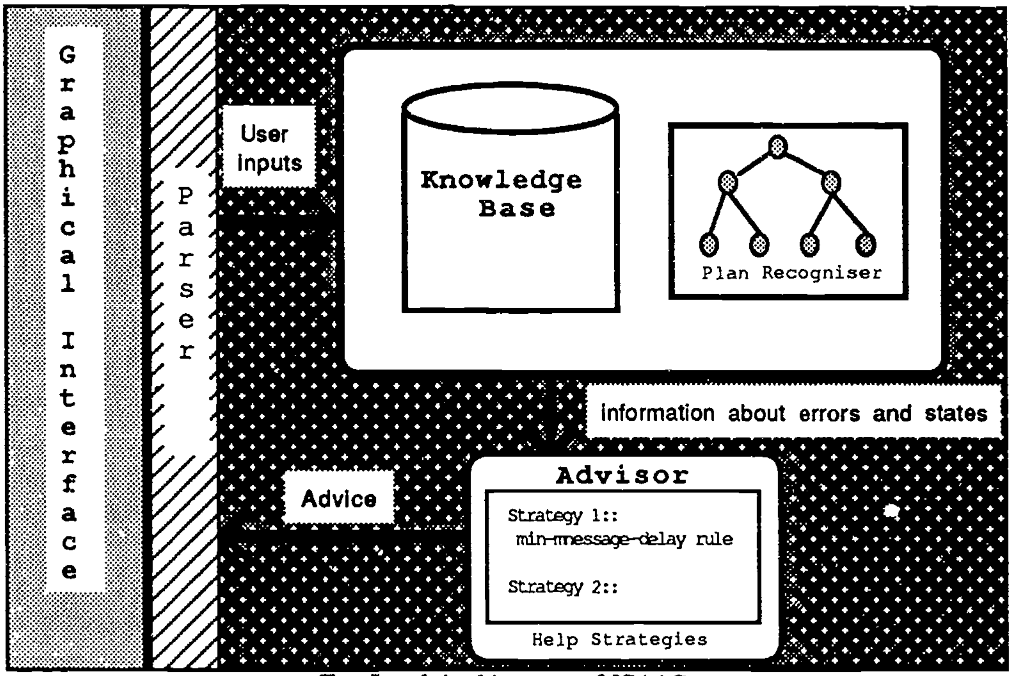
The conceptual design provides the necessary framework on which the detailed design can be built. This conceptual design gives a high-level, abstract models of both the architecture and the working of the system.

Knowledge Representation of ISAAC

In summary, the use of objects have enabled a natural means of modelling of information. Object-oriented represented not only provide a flexible means by which information related to an abstract/physical item to be easily attached to the item, but it also allow other procedures such as those constraints to be attached easily. With the capability to incorporate procedures such as constraints allow automatic checking to be carried out. This is both a natural way of incorporating the instructional design model together with the tasks, as well as, a convenient way of dealing with checks and controls. Finally, this knowledge representation as seen in the prototype provides the basic framework on which other parts of the instructional design knowledge could be added in the future system.

Architecture

The architecture of the prototype version of ISAAC retains most of the features as the initial conceptual model. The diagram below depicts the top-level architecture of the prototype.



Top-Level Architecture of ISAAC prototype

In ISAAC, graphical interface is used for the writing of all the inputs such as the instructional goal and the instructional components. In this interface, there are simple graphic tools where the user can create nodes [goals/components] as well as links [relationships] between nodes. He/she may delete any node or link at any point of time. Direct manipulation interaction style is employed in the user interface. The parser is the same as that described in the conceptual design. There are several ways to design this plan recogniser. The first way [using pre-stored plans] demands that all the possible plan models be defined. It is not possible in the context of designing an instructional design, as there are numerous number of possible designs and variations. The second [using DCG rules] and the third option is using production rules . One difficulty is to design ways to check for violations of the instructional theory. The last option [using constraints] was adopted because it provides a more natural means to design the plan recognition. Conceptually, the plan recogniser and the knowledge base are separate entities. However, since constraints are used to monitor the deviations and trigger the advice generator, the constraint mechanism essentially functions as the plan recogniser. Therefore, because of this representation, the "plan recogniser" [the constraint mechanism] are part of the knowledge base. The advice generator remains as a separate entity as described in the conceptual design. The advice generator uses several strategies. The strategies used are:

- use of common terminologies and technical or instructional design theory - jargon is avoided,
- provide two levels of advice,
- one level corresponds to the procedural aspect of the task,
- another level presents the conceptual level of the task.

In ISAAC, the advisor will always present the procedural level of advice to the designer. Only when the user wishes to read the conceptual level explanation, it will be presented. "Canned text" is used for all the advice and explanation. Therefore, instead of actually generating statements of advice for the user, as suggested in the conceptual design, the advice would search for the appropriate text for the display. However, the advice generator does some

361

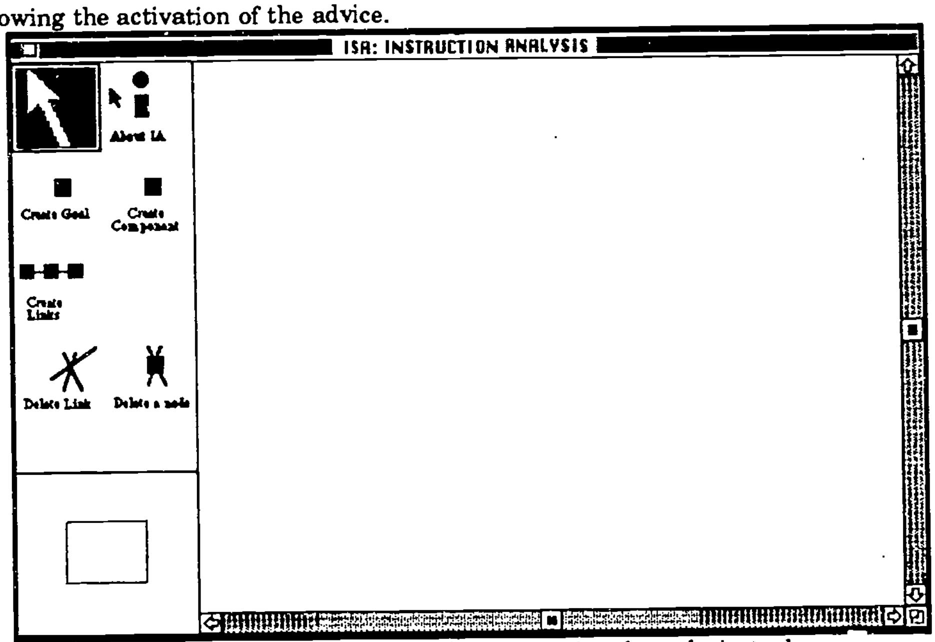
contextualisation of the text to suit the actual context which the user is working under. For example, it extracts certain information such as the names of the objects involved and the type of errors committed and add these information into the advice. However, it does not take into account of the states as mentioned in the previous conceptual design. This advice generator, is therefore, a simplified version of the conceptual advice generator.

The User Interface Design

An attempt is made to model the designer's normal way of carrying out instructional design while, at the same time, incorporate the instructional theory together in the model. This is extremely difficult and tricky. There are instances when the two conflict. For example, as mentioned earlier in the requirement analysis, one intuitive way by which the designer works is not to write the instructional objective explicitly. This conflicts with the theory that demands the explicit recording and careful planning of the objective. Therefore, one strategy is employed: that is, the instructional theory will always override the user's way of carrying out the instructional design. This is chosen since ISAAC is model-based. The previously stated specification are all fulfilled in the interface design. The following sub-sections will present the overall presentation of the interface [giving examples of sample screens] and the aspect of designing the dialogue control and flow.

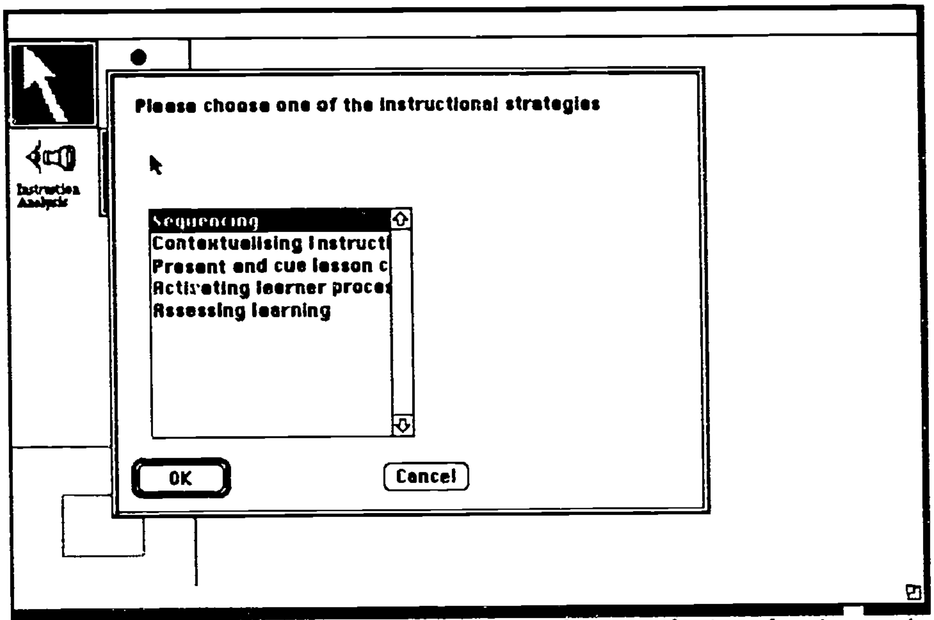
Presentation/Screen Design

The presentation of ISAAC has been designed to match those of the Macintosh™ user interface. It will provide a familiar working environment for users who already have experience with the Macintosh™. For the novice users, the user interface will not pose a problem as the interaction style does not, as far as possible, require the users to remember commands or complex key strokes. The point-and-click form of interaction is relatively easy to learn and apply. The following diagrams present sample screens from the system. Presented are the windows [or contexts] which the user will be working in, as well as a screen showing the activation of the advice.

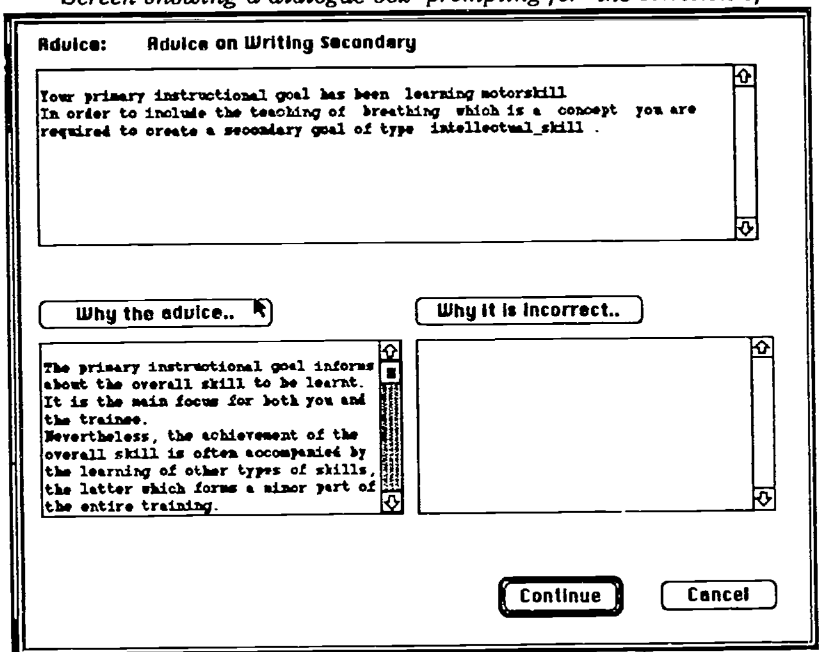


Interface for instructional analysis task.

Note the the screen for the instructional analysis, on the left-hand side of the window is the tool panel where the designer can select tools for the creation/deletion of the instructional goal.



Screen showing a dialogue box prompting for the selection of an instructional strategy

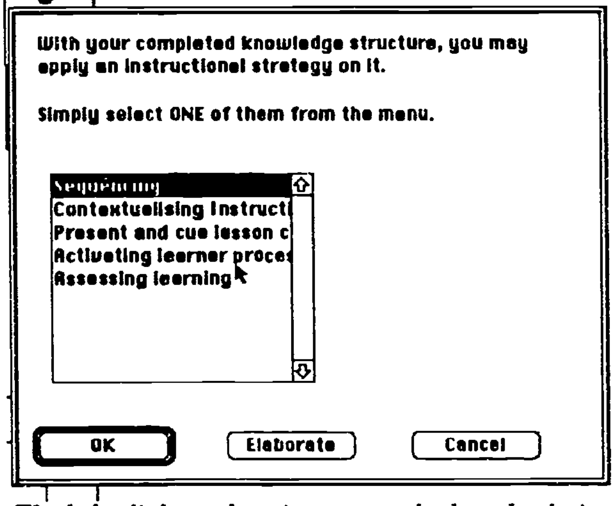


Dialogue box for the advice.

The figure above shows the advice dialogue box, where it shows the two option buttons, "Why the advice..." and the "Why it is incorrect...". These two options give the designer more information regarding the advice given.

Instructional Strategies

The task that involves the application of instructional strategies are evoked by the selection of the "Instructional Strategies" tool in the "ISAAC MAIN" window or the menu option, "Instructional Strategies", from the "Go To ..." menu. Upon entry into this task, the designer will be prompted to select one instructional strategy to apply. Subsequently, the user is presented a help dialogue box to guide the him/her in the procedure of sequencing. It is desirable to give the user an option to set this help on or off for the next entry into this task. However, for the purpose of the prototype, this is not implemented.

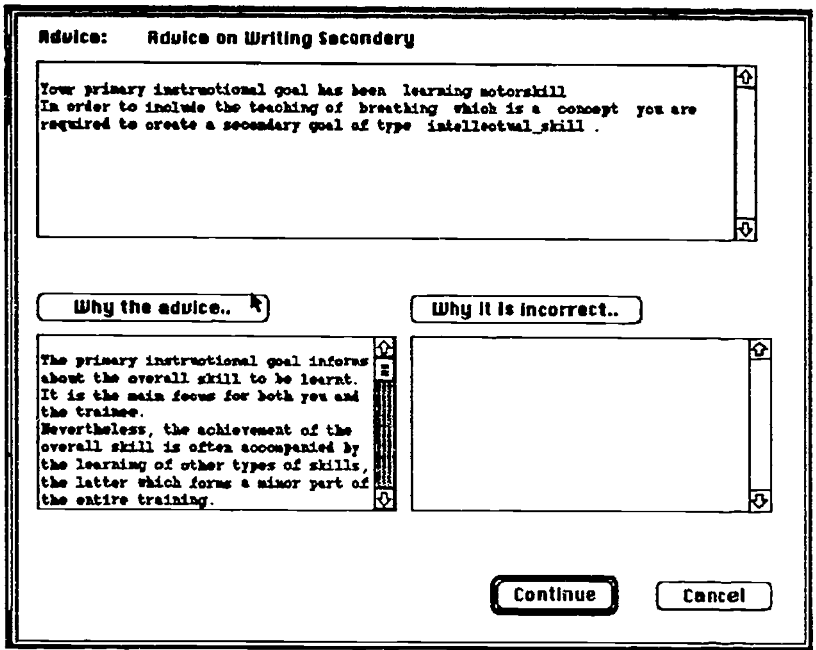


The help dialogue box is presented when the designer selects the "instructional Strategies" tool.

Once again, the designer can make use of the tools presented in the tool panel to sequence his/her knowledge structure. As the user sequences the components, the components are written into another window, the "Sequencing List" window. This window serves to record what the designer has done in text form. This means that the designer will be able to print the sequenced instructions onto a hard copy for future usage and reference.

Advice-giving

The advice-giving is evoked when the system detects a deviation in the user's action from the instructional design principles. Once the advice is called, it will present a dialogue box which, upon entry, shows only the procedural aspect of the advice. By procedural, it refers to the actions which the designer may take in order to correct his/her mistakes. It does not include explanations of why the procedures are recommended and what theoretical basis these procedures are based.



The advice dialogue box, showing the options for further explanations to be displayed

For explanations as to why the advice is given and what theoretical explanations, the designer/ user will have to select the buttons provided in the dialogue box. The designer can exit from the advice mode by selecting the "Continue" button and He/she can resume his/her previous task or make the necessary amendments.

No details are provided here of the implementation of ISAAC. It has been built using MacProlog™ and its Flex™ sub-system. (for details , see the MSc thesis by Lai-Mui Ching)

EVALUATION

Evaluation of ISAAC used experts [university teachers] and novices [postgraduate students]. The approach has been to take a critical HCI approach to the task. Full details are not given here however, our evaluation has discovered some interesting but expected outcomes. [6] Some of the findings have been encouraging. For example, generally, the interface has been easy to use. The aspect which are "problem areas" is mainly due to two aspects: the notion of an active advisor and the instructional design model. The former is still a real research subject; the interesting finding is that the subjects do not really understand the role of an advisor, which has been assumed otherwise in the literature. This could be be attributed to the advisor being a machine, not a human being. The employment of one instructional design model in the system, as expected, did pose some acceptance problems. Unless the users are fully convinced of the model or theory, they are hesitant in accepting and following the advice.

Issues and Problems

Amidst of the entire design and development process, there are numerous problems and issues. Some of them have been solved, one way or another; some of them, however, are either beyond the scope and time constraint of the project or beyond the capability of the software/platform to handle.

These issues and problems can broadly be classified into four categories:

- instructional design theory
- the current state-of-the-art of advice-giving

---

6 See the MSc thesis by Lai-Mui Ching, Heriot-Watt University, 1992, " ISAAC: **An Instructional Design Advisor for Computer-based Learning**

•environment/language for the development of the prototype
•technical issues and constraints

Weakness of the Gagné-Briggs Model

Instructional theories are not without weakness. Like many other theories, the Gagné-Briggs model lacks rigorous testing. [Andrews et al, 1991]. The instructional design model proposed is based on a number of assumptions, many of which are not tested or accepted by the experimental subjects. In addition, there is still not thorough understanding of all the learning outcomes described by Gagné. Out of the five learning outcomes, only the intellectual skill is widely accepted.

Current State-of-the-Art of Advice-Giving

There is still a serious lack of empirical testing and validation of the advice approaches used by ISAAC. There is also the problem of the current technology in implementing advice strategies. This is a multiple-disciplined area of interest. There is, at present, no simple way of implementing a context-sensitive advisory system, especially in the active type of advice-giving facility. The present state-of-the-art of plan recognition and natural language understanding have provided the ultimate solution to the understanding of user's actions and plans.

Technical Issues and Constraints

Most of the technical issues and constraints arise from the usage of MacProlog™ and its Flex™ sub-system as implementation tools. Some of the problems arise because the Flex™ sub-system, for example, it is not a fully object-oriented system. The frame sub-system gives an additional feature to the production rules system, where the latter is more prominent and complete in the Flex™ system. The few issues encountered include the following:

•the Flex™ system does not provide the full environment for the handling of objects and constraints. The whole structure for object representation can be considered to be basic. For example, constraints that are defined for a frame [equivalent in concept to a class of an object system] are not inheritable in its sub-frames or its instances. This result in the duplication of efforts in trying to define constraints for each instance of the frame.

•there is no support in the saving and loading of the frames and instances that are defined by the users. This is an important feature [the saving and loading of the designer's work] to have but in order to achieve it, an explicit routine has to be written to capture all the dynamically created instances, together with their attributes and constraints. At the same time, the graphic objects on screen are not linked directly to their corresponding instances defined. This routine not only have to capture all these information, but it also have to write these information in a form which can be parsed when it is loaded. It has been a very time consuming task to implement this routine.

CONCLUSIONS

The prototype of ISAAC has been developed to demonstrate the possibility of introducing an advisor into a computer-based instruction design system. Despite its limited scope and capabilities, this prototype system carries within itself, a highly useful, feasible and interesting structure that can be used to produce a system useful to instructional designers at large.

Extension to the Present Knowledge Base

The present knowledge base can be extended to include the rest of the instructional model which have not been included yet. The object representation can be used for representing these knowledge in the same way as the rest of the knowledge. It must be mentioned here that the representation of the rest of the knowledge was not the most difficult task. The most difficult task was the extraction of the knowledge from the instructional design theories.

Shell to other instructional design theories

The system ISAAC has been designed using the similar idea as that of an expert system, where the knowledge base is separated from the inference engine. Here in ISAAC, the knowledge base, which contains the frame definitions [together with its attribute and constraints] can be replaced by another knowledge base containing a totally new set of knowledge. The amount of effort that will be involved to replace the present knowledge by another knowledge will be small. The only consideration perhaps, will lie in interface. The input should be graphical inputs

From frames to objects

Perhaps, one improvement for ISAAC's knowledge representation could be to transfer the present Flex™ frame structure to a true object-oriented environment, like the Prolog++ or KEE [Knowledge Engineering Environment], since the Flex™ system does not provide a complete object-oriented environment. With a complete object-oriented environment, some of the constraints which are faced when using the frame sub-system of Flex™ can be overcome [for example, the support for dynamically created objects and their instances].

Extending the Advice-Giving

There are many ways in which the present advice-giving capability can be extended:

Batch Mode

Beside having an active advisor trailing the path of a designer and giving advice along the way, a batch mode feature could be added. In batch mode the designer may switch the advisor off and carry out his/her design. Later, he/she may get the advisor [which is switched off] to "compile" the design. This compilation will produce any comments and recommendations, just like those produced by the active advisor. Batch mode has the advantage where the designer is not interrupted by the advisor during his/her design work, and at the same time, He/she can get some feedback on the design that is constructed. The disadvantage of this feature would be the designer may be tempted to finish the entire design before allowing the advisor to "check" it. This could result in enormous amount of comments or recommendations which could be discouraging to the designer, especially if He/she is still a novice in this area. At the same time, with so much comments to assimilate and so many corrections to make, it may not be an effective way of giving advice.

Multimedia and Hypermedia Inclusion

In the present advice, all the advice messages and the explanation messages are textual. This could be extended to include the use of new technologies such as the multimedia and hypermedia capabilities. In the simplest form, a static graphic could be used to illustrate templates of instructional design plan. Hot-spots could be used on individual words, phrases or chunks of text in order to provide linkage to related references or text or even graphics. This is, in effect, using the hypertext and hypermedia capabilities in the presentation of the advice. This will be a useful feature, especially for those who wants to explore instructional designing through the use of the advice facility. It will be a more effective learning tool than what it presently can offer. In addition to the above capabilities, multimedia capabilities could also be incorporated. Animated or motion picture clips could be used to demonstrate the examples given in the advice more effectively. Therefore, any extension to the presentation of the advice will greatly enhance the capability of the advisor for the purpose of learning and exploration. Complex features like those of multimedia will be useful to have if there is a high priority to employ the advisor in the teaching and learning instructional design.

What-Ifs

This is a feature which is similar to many simulation programs. It would allow the designer to "try" different alternatives and obtain feedback immediately. This will enable the designers to learn about the application of the theory by trying to apply them and getting "corrections" after each try.

Dynamic Generation of Advice

This is an extremely useful capability. Using of canned text restricts the context-sensitivity of the advice content. The advisor will be able to give better, higher context-sensitive recommendations and corrections if it has the capability to generate advice dynamically for a particular context. This capability is, unfortunately, an entire research subject on its own and must await further developments.

Natural Language

Natural language will be employed especially in the dynamically generated advice. As the system generates an advice, it will have to formulate the advice in such a way that it is grammatically and pragmatically correct [assuming the natural language refers to the English language]. Natural language capability can be used to extend the advisor to be able accept query from the designers in the form of a natural language sentence and to generate an advisor in the similar form.

Summary

ERIC

The idea of having a computer-based authoring system running with an on-line active advisor has been demonstrated in the ISAAC. Despite being a prototype system, it carries within itself a vision about the future authoring tools. Its structure and design can be extended to become a complete, usable and powerful authoring system. Alternatively, the idea that ISAAC has can be incorporated into existing authoring environments to enhance their capability in the pedagogy level of support. It must be noted that the solution of providing an advisory system in a specialised context, can be extended in many ways. There is no one simple, direct solution, which one may be tempted suggest. This is so partly due to the complexity that is involved in the pedagogy aspect of instructional design, and more so due to the need to develop a system that is usable, extensible and versatile system.

There is a place for an advisor in the context of the instructional design. ISAAC has demonstrated both its feasibility and the related issues. Issues, like those of the advice-giving aspect, are inevitable. While still recognising the existence of these issues, the idea proposed in this project can be extended using whatever techniques that are available into a usable system.

References

Andrews, D.H & Goodson, L.A. 1991, *A Comparative Analysis of Models of Instructional Design.* Anglin, Gary J. [Ed]. 1991. *Instructional Technology: Past, Present and Future.* Librarians Unlimited Inc.

Briggs, L.J. [Ed]. 1977. *Instructional design: Principles and applications. Englewood* Cliffs, NJ: Educational Technology Publications.

Carroll, J. M. et al 1987. The Minimum Manual, *Human Computer Interaction, 3*, 123-153.

Carroll, J.M. & McKendree. 1987. Interface *Design Issues for advice-giving expert systems.* Communications of the ACM, Vol. 30.

Carroll, J.M. & Carrither, C. 1984. *Training Wheels in a User Interface.* Communications ACM, 27 [8].

Carroll, JM & Aaronson, A. 1988. Learning *by Doing with Simulated Intelligent Help.* Communications of the ACM, Vol. 31, pg 1064-1079.

Coombs, M.J. & Alty, J.L. 1984. *Expert systems: An alternate paradigm.* Int. J. Man-Machine Studies, 20.

Dick, W & Carey, L. 1990. *The systematic design of instruction.* 3rd ed. Scott, Foresman & Co.

Fikes, R.E. 1981. *Automating the Problem Solving in Procedural Office Work.* AFIPS Office Automation conference, Houston, TX.

Fitter, M. 1979. *Towards more "Natural" Interactive Systems.* Int. J. Man-Machine Studies, 11.

Fischer, G, Lemke, A and Schwab, T. 1985. *Knowledge-based Help System, CHI* '85 Proceedings.

Gagné, Roberts. 1985. *The conditions of learning.* 4th Ed. New York: Holt, Rinehart and Winston.

Gagné, R.. 1991. *Analysis of Objectives.* Briggs, L.J. & Gustafosn, Tillman. Instructional Design: Principles and Applications. Englewood Cliffs, NJ: Educational Technology Publications.

Gagné, R & Briggs L & Wager W. 1992. *Principles of Instructional Design.* Library of Congress Cataloguing-in-Publication Data.

Hannum, W. 1988. *Designing Courseware to fit subject matter structure.* Jonassen, D.H. [Ed]. Instructional Designs For Microcomputer Courseware. Lawrence Erlbarum Associates, Inc.

Hobbs, D. J. 1990. Second-level design of a knowledge based educational advisor, *Educational Training Technology International*, 27,2,216-223.

IDE Reference Manual. Xerox Corporation. 1988.

Jonassen, D, Harris, N. 1991 Analysing *and Selecting Instructional Strategies and Tactics.* Performance Improvement Quarterly, Vol. 4.

Jonassen, D & Hannum, W & Tessmer, M. [in press]. *Handbook of task analysis procedures.* New York: Praeger.

Jones, M. K. Li, Z. and Merrill, M. D. 1990, Domain Knowledge Representation for Instructional Design. *Educational Technology*, October.

Korionos, A. 1991. The development of an instructional design expert advisory system (IDEAS), Interactive Learning International, 7,3,259-260.

Lloyds, E. 1990. *How Learner Characteristics can mediate the effects of giving conceptual details during training.* Human-Computer Interaction - INTERACT '90.

Merrill, M.D. Winter 1987. *Prescriptions for an Authoring System.* Journal of Computer-Based Instruction,

Vol. 14 [1].

Rich, E. 1983. *Users are Individuals: Individualising User Models.* Int. J. man-Machine Studies, 18.

Saule, M.R. n.d.

Type H for Help: On screen Help as a Teaching Tool

Spiro, R.J., Feltovich, P.L., Jacobson, M.J. & Coulson, R.L., 1991 Cognitive flexibility, constructivism, and hypertext: random access instruction for advanced knowledge acquisition in ill-structured domains, *Educational Technology* , 31,5, 24-33

Shrager, J, & Finin, T. 1982. *An Expert System that Volunteers Advice*, AAAI.

Sleeman, D. and Brown, J. S. (Eds) *Intelligent Tutoring Systems*, Academic Press.

Stevens, A & Collins, A. & Goldin, S.E. 1982. *Misconceptions in Students' Understanding.* Intelligent Tutoring Systems. Sleeman, D. & Brown, J.S. ed. Academic Press, New York.

Venezky & Richard & Osin. 1991. The *Intelligent Design of Computer-Based Instruction.* Longman Publishing Group.

Whitaker, Elizabeth T. & Bonnell, Ronald, D. Vol 1, No2, 1990. *Plan Recognition in Intelligent Tutoring Systems, Intelligent* Tutoring Media.

Wilson, Brent G. & Jonassen, David H. *Automated Instructional Systems Design: A Review of `Prototype Systems.* Journal of Artificial Intelligence in Education, Vol2 Winter 90/91.

Computers and Visual Databases: Designing Open systems for Flexible Image Cataloguing

G.F. McLean V. Wyatt
Mechanical Eng. History in Art
University of Victoria
Victoria, British Columbia, Canada

In Art History, large quantities of image data are stored and referenced in 'slide libraries'. These libraries are central repositories of image collections that form the basis of Art History research, and that can be accessed when teaching about art and its historical context. Just as physical laboratories form the essential backbone of a technical education so the slide library can form the essential framework from which Fine Arts educators build their program. In this paper we describe the design of a system which has been developed to provide new ways of associating contextual historical information with images. The system is based upon flexible, dynamic linking between database objects presented to users in an entirely visual way.

Universities worldwide have a resource of enormous potential in the form of large numbers of catalogued 35mm slides. However, the prevalence of 35mm emulsion-type slides leads to an array of technical and logistical difficulties which detract from the usefulness of this resource. The frailty of the slides and the ease with which slide projectors damage them require that the slides be treated as rare commodities rather than available resources. Therefore, allowing access to slides must always be a compromise between opening the holdings for unrestricted viewing that could damage the slide, and protecting the holdings for their archival value. Apart from preserving the physical quality of the slide, the technology poses other, more profound, barriers to teaching (Wyatt, 1993):

- The existence of only single copies of slides makes student review difficult.
- Images can only be compared using an expensive viewing room containing multiple slide projectors.
- In a classroom setting, an instructor must pre-determine the set of slides that will be shown, in a roughly sequential order, thus reinforcing the power relationship between professor and student, and certainly not encouraging any sort of spontaneous classroom discussion.

The introduction of digital imaging technology and hyper-media systems to access these image data promises to change this status quo access to images:

- A digital version of a slide can be copied easily without loss of quality.
- It will never decay.
- It can be shared between users who are remote in space and time.

- The digital version can be logically associated with a rich array of catalogue or other descriptive data.

Not only does the use of digital imaging technology and hyper-media suggest an obvious transposition of the slide library from being film-based to being image-based, its introduction suggests technical and logical advances which can serve to radically change the role of visual images in arts education. Whereas the slide itself was previously inseparable from the image contained in the slide, the use of hyper-media makes the image independent of the medium on which it is displayed. Thus the image becomes safe to use, ubiquitously available, easy to obtain and inexpensively propagated. In short, the transition to hyper-media based systems using digital imaging breaks down the barriers caused by conventional technologies that limit the utility of the slide library.

The system described in this paper introduces a new and subtly different concept concerning the organization and retrieval of images. Whereas some may view the computer slide library as simply a fancy form of slide projector (therefore carrying with it all of the limitations in learning etc. posed by the slide projector), we wanted to design a system which would 'draw people in', encourage them to explore, to share ideas, to 'own' the images. With this system, libraries, special interest groups and individuals can all create create their own catalogue of the image database. These catalogues can be shared and continually updated as new insights into the image data are gained. In addition, users may independently define visual regions (i.e. buttons) within images to allow for the generation of visual database querying, further breaking down the barrier between visual data and text-based associations. Finally, the system is 'open' and horizontally organized. Images may be added to the catalogue, new fields defined or new catalogues created without any requirement to interrupt the system operation or rebuild any organizational structures.

In the remainder of this paper the fundamental concepts underpinning our design are outlined, and examples provided which illustrate the pertinent points of the system's design and operation. The paper concludes with a description of new work in image processing and computational vision being planned to further improve the manner in which visual images can be integrated into Fine Arts education.

Defining Image Attributes

The status quo in cataloguing images is to associate each image with a catalogue name and a number of associated attributes. An image is classified according to its origin, media type, physical location (i.e. museum collections) etc. In the traditional view, these classifications are defined as attributes of the image itself; they form secondary data which supports the primary information of the image.

With and electronic slide catalogue the image is no longer a scarce resource: multiple copies can be viewed simultaneously and the image can correctly 'belong' to many classifications according to concepts of ownership (special collections, discovered themes, course materials etc.) or concepts of shared properties (20th Century, Northwest Coast, Northern Form-line, Haida etc.). Moreover, these associations can be classified into those that are superficially known when the image is accessed, those which are discovered through research, those which are created for special purposes by users and those which are created temporarily for a particular use. This richness in association between an image and its contexts cannot be captured using conventional, linear cataloguing methods. Recognizing the need to maintain and promote these associations has two direct implications for systems design.

1. It implies that users ought to be allowed to create their own associations and definitions of significant image categories or ownerships.

2. It suggests that the system ought to be based upon the idea of associating images with attributes in a flexible and modifiable manner.

Research and exploration are all about defining new meanings, discovering new associations, destroying useless or outmoded categories and creating new ones that will help to clarify the thoughts of a student or researcher. Therefore, although an image will enter the system with a set of meanings and classifications, our hope is that the system will allow new meanings and associations to arise, as researchers learn more and modify their needs. In our system we wish all associations (links) between images to be negotiable. It will be the inherent right of each user to define their own unique set of meanings and associations. Furthermore, we wish users to have access to each other's meanings, to be able to act in groups. Finally, we wish to continue to allow knowledgeable scholars to guide this experience, through a system in which many levels of association are created.

Object-Oriented Cataloguing

The preceding discussion of the relationship between an image and its defined context suggests that the image itself is relatively unimportant as far as the cataloguing goes. Of course, without the image there is nothing, but similarly, without the definitions of the 'essence' of the image, it merely becomes a puzzle. The image requires context to be of any utility, but the real power of the system comes in allowing users to devalue existing contexts by creating new ones. Therefore, it is the context of the image which becomes most important; the image is merely an attribute of contextual objects.

To this end, we define a 'catalogue object' as the primary component of a library. A catalogue object defines a set of logical associations between a description of defined contexts, an image, a visual icon, and a set of image regions of interest. These regions of interest define visual areas which, when selected by a user, will direct the system to perform certain actions, such as loading another image or performing a database search on a specific theme. The structure of the object is flexible, allowing associations with other media extensions, such as sound and video, as these become available.

In contrast to the traditional slide library approach, which rigidly associates images with their contextual definitions, the use of image catalogue objects makes no formal link between an image and any defined context. A comparison of the two approaches is illustrated schematically in figure 1. An image may now be associated with an infinite number of different catalogue objects, each of which may define a potentially infinite number of different contexts for the image. In addition, there is no longer any necessity for images and their contextual definitions to be physically associated: it becomes possible to include remotely located images in a locally defined library of catalogue objects.

Using the catalogue object approach allows the slide library to be organized in many different ways. For example, a single object can be associated with an image to define all possible contextual associations with the image. This approach presents a simple extension of the existing slide library concept into the computer, taking advantage only of the improved access to display facilities. In another approach, which can be implemented in parallel with the first, multiple objects corresponding to the diverse cataloguing ideas of multiple users can be associated with a single image, to create a special collection, or to meet the particular

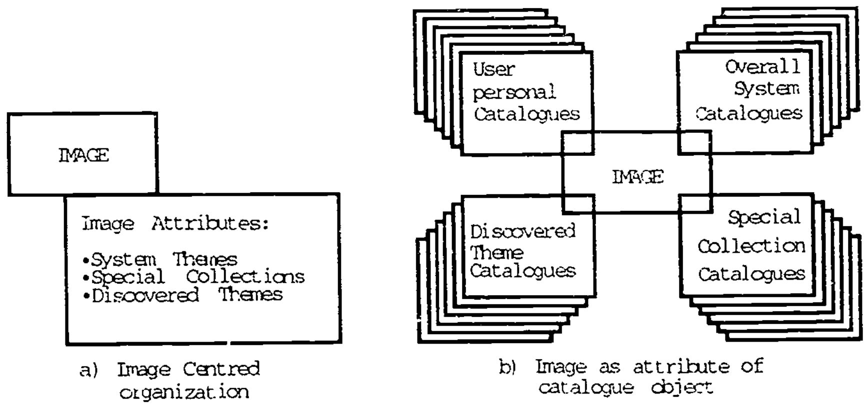


Figure 1: A comparison of cataloguing ideologies and their relationship to image data.

research interests of a graduate student. The image is now an important attribute of the object definition, not the central piece of information around which the system is organized.

Catalogue objects may be organized and searched according to any pattern. For example, users may perform alphabetical searches, keyword searches or any format-free database search. The key is that the object includes particular associations attributed to an image. Keeping in mind that because the same image may be identified by multiple objects significant operational power is gained. Consider, for example, the possibilities suggested by creating objects using three levels of object definition:

- A 'system' object which includes the superficial catalogue contexts.
- A 'special collection' object, which identifies the image as being of interest to a particular group or project.
- A 'personal catalogue' object which associates images with uniquely personal attributes defined by individual users.

A user can simultaneously access all three levels of object definition, de-constructing the boundaries created by any single levels interpretation of the image and encouraging the creation of new associations.

The object definition approach relegates the actual image to being merely an attribute of the defined objects. However, from the users point of view the image is still the dominant feature of the system: all database interaction is performed by viewing images, all system response to the user is provided in the form of images and all system actions are initiated by 'selecting' visually defined regions with a pointing device

The object definition includes an association with an image, and with a thumbnail version of the image which provides a visual description of the image in compact form. The system presents the catalogue objects by displaying palettes of these small thumbnail images. Therefore, all interaction with the system is performed with visual images as the primary attribute; the text-based contextual description is the secondary attribute. From the user's point of view, image data is paramount and the contextual definition secondary, from the systems point of view, the contextual definition is primary while the image data

ERIC

is secondary. The chief benefit of this approach is that an apparently visual cataloguing system is achieved by a very simple association of text and image data.

Hypervip

Hypervip is a software system which has been developed at the University of Victoria to implement the object based cataloguing concept described in the previous section. The software has been written in the C language running under the Unix operating system with X windows graphical interface (Openwin). As such, Hypervip is similar to a variety of other dedicated applications of hypermedia technology, as referenced in (Lamb, 1991). The unique feature of Hypervip is its total dependence upon the user's navigation through the database using visual interaction.

Hypervip consists principally of two programs and associated utilities. The first program displays the image associated with an object, then manages events. The second program queries objects and performs database searches. Object definitions, context descriptions, images and image icons may all exist in different places, as defined by environmental variables. Moreover, multiple locations of each object attribute can be defined, allowing users to tap into various levels of system cataloguing.

Loading an object causes the system to locate the image associated with the object and to display it. The image display monitors mouse activity so that when a mouse button is pressed in one of the pre-defined regions of interest the associated action command is issued. The regions of interest operate in a similar fashion to 'buttons' in a hyper-text stack (Jonassen, 1989). The difference is that a hyper-text button usually invokes the display of another pre-defined object (a link). This may be done in Hypervip by associating a load command with a button. However, Hypervip allows the execution of any system command when a button is pressed, including the generation of a database search command.

A database search command instructs the system to locate all objects with a particular set of desired attributes. The search locates the descriptions associated with all objects in the search path and these descriptions are queried according to the database search. The collection of objects responding positively to the search are grouped together to create a temporary object which displays a 'palette' of thumbnail images of each object in the set. This provides a visual overview of the response to the search from which particular images can be selected. In addition to creating the overview palette, the search program also creates a set of button definitions, so that selecting any one of the small thumbnail images with the mouse will invoke a load command for that object.

The load program defines utilities to create button definitions and the specification of the desired actions. It can also modify the contextual descriptions of each object. In this manner, a user is empowered to modify the linkages and classifications associated with each image while the system is in operation. The system requires no re-compiling or re-building to accomplish this. Similarly, the addition of a new image into the overall system entails no management overhead, as the new image is automatically included in all database searches by virtue of its association with an object located in one of the object search paths.

An example of a Hypervip Palette is shown in figure 2. In this case, a collection of thumbnail images with the similar theme of 'SW.Acoma' have been located and displayed. Each image in the palette defines a visual button: selecting one of the displayed thumbnail images causes a high resolution version of the image to be displayed, as in figure 3, which in this case contains a variety of visual themes. Regions of interest - which have been defined as buttons - are shown with their bounding rectangles hi-lighted. Selecting the image of the

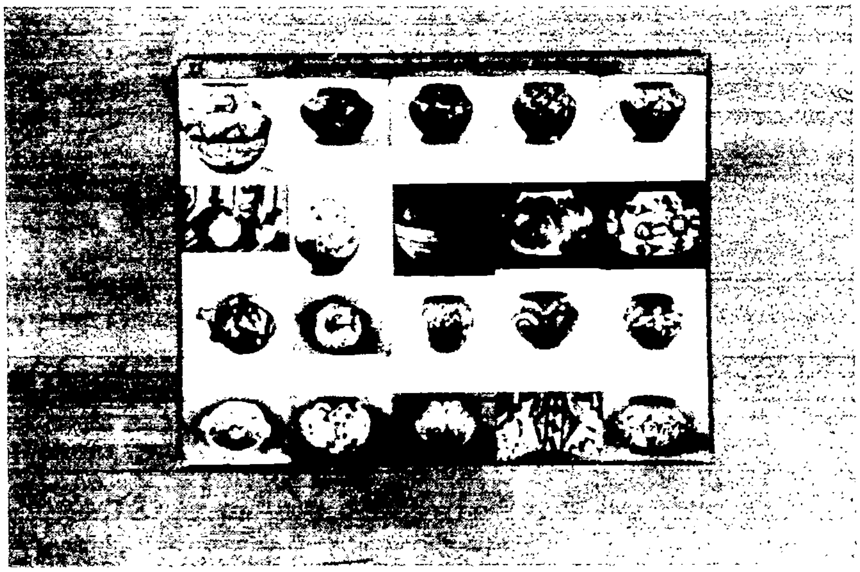


Figure 2: A palette of thumbnail images created by a database search

Figure 3: An image displayed by selecting a thumbnail from the palette. Visual buttons are defined by rectangular hi-lighted regions.

BEST COPY AVAILABLE

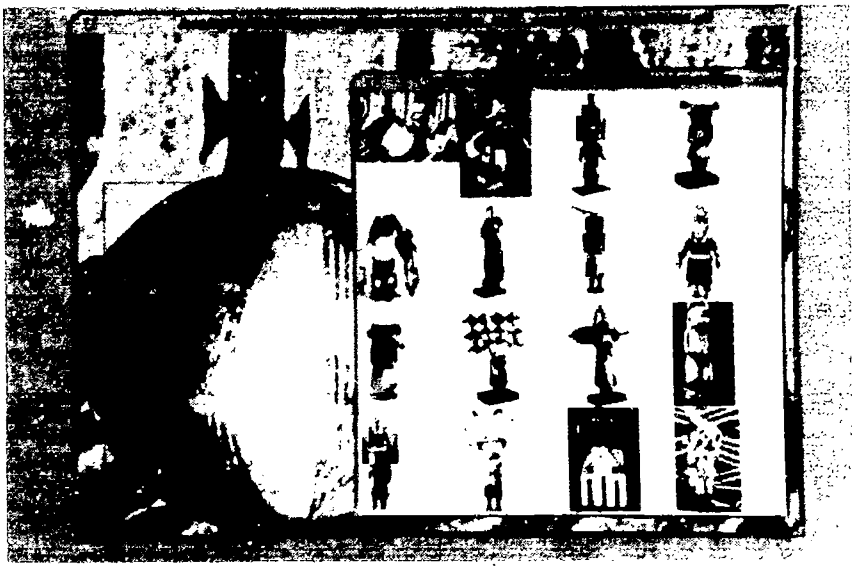


Figure 4: A new palette of images created by selecting the visual region associated with the doll in the previous image.

doll causes a database search on the theme of 'Kachina Doll', initiating a search for these images, the creation and display of the palette of Doll images, shown in figure 4. Finally, several Kachina Doll images may be selected from the palette and displayed simultaneously to allow direct comparison as shown in figure 5. The example portrayed by these images illustrates the operation of Hypervip, and also illustrates the power of the system by allowing the association of textual descriptions (for database searching) with visual themes. In this manner, the initial display of a set of images led directly to the comparison of a set of dolls. The user required no knowledge of keywords, and was able to pursue visually interesting themes. In this example alone, which represents less than 1 minute of interactive work on a workstation, the user reviewed over 60 images superficially and four images in detail. The user could have defined new contextual definitions for any of these images at any time. Similarly, the addition of new images to the database would not require any rebuilding of links in order to be included in the users search, as all links are dynamic, horizontal and open.

Future Plans

The Hypervip project at the University of Victoria is part of a larger investigation into the use of electronic systems in Fine Arts education. Providing a flexible and easy to use method of cataloguing images has been the primary purpose motivating the development of Hypervip. The system achieves these goals nicely. As the project progresses, we are beginning to investigate ways in which automated image enhancement and analysis can be used to aid in the research aspects of our collections. For example, we are developing methods for image enhancement which, while providing methods to improve the visual quality of the collection, will also provide the opportunity to restore historical photographs of art, thus allowing direct comparison with contemporary art. In related work, we are investigating the use of the computer to analyse the form of the images, defining similarities of form as database attributes. In this manner, Hypervip will be used to direct the researcher to literally 'discover' classes of 'shared essence' in the database that could never be identified using manual methods of common feature identification.

ERIC

Figure 5: A comparison of three images chosen from the previous palette.

Conclusions

In this paper we have discussed the requirement for flexible, open ended systems which can allow visual images to be effectively used in educational environments. Much of the design consideration has been motivated by the need to develop systems which do not impose arbitrary boundaries or definitions on the image data, but allow users to create their own ideas regarding the organization and meaning of images. In developing the new system ideas, it was paramount that the many technical and logistical boundaries created by the use of 35mm slide image formats be destroyed.

The proposed system concept goes far beyond a simple electronic extension of the slide library. Rather, it recognizes the right of individual users to access the image data according to their own style, to add their own meaning and to impose then own structure. We think of the existing slide library as a sort of 'closed stacks' library - the user is allowed access to the superficial catalogue data and has borrowing privileges but is never allowed to browse. Our first goal is to create an 'open stack' library in which the user can browse through the collection in any desired order. Finally, by virtue only of the electronic media, we wish to allow the user to define their own collection, replete with associations, definitions and boundaries that are personally defined and owned.

The development of Hypervip represents the first concrete association between researchers involved in imaging technology and Fine Arts at the university of Victoria. The scope of applications involved by Fine Arts is vast. This project is serving to identify a platform from which further innovations in multi-media education can be developed.

References

D. Jonassen (1989) *Hypertext/Hypermedia.* Educational Technology Publications. Englewood Cliffs, NJ.

A.C. Lamb (1991) *Emerging Technologies and Instruction: Hypertext, Hypermedia and Interactive Multimedia.* Educational Technology Publications. Englewood Cliffs, NJ,

V. Wyatt and G. McLean (1993) Imaging databases in research and teaching: global perspective and new research technologies. *Visual Resources*, To Appear.

Multimedia and the Teaching of Literature

Carla Meskill and Karen Swan
University at Albany, State University of New York

Imagine a tool, a database of information, whose use promotes and supports rich conversation around literary works; a tool that helps students make connections, build meaning and articulate their thinking. Imagine a class of twenty-seven high school students assigned to read Shakespeare for the first time. In the front of the classroom is a large video monitor which displays a vivid color photograph of pastoral Stratford-on-Avon. In the course of facilitating a discussion that sets the scene for the play, their teacher selects many additional still images and video sequences from a computer-generated menu. These cue and serve to illustrate students' discourse. They are perhaps scenes that depict the life and times of the playwright, the *mise en scene* for a variety of stage productions, thematic works of art, segments from today's media and sample treatments of the work in contemporary film. Two days later, small groups of students are seated in front of a smaller version of this tool engaged in lively talk. One member clicks the mouse to access video, audio, text and graphics which they work to relate to the play they are in the process of reading. Their task is to select and assemble such elements into a class presentation concerning an aspect of the piece. If the group becomes curious or puzzled, elaborated explications are accessed on the screen and discussed, interpretations and corresponding supporting materials are negotiated. Four days later, once the small groups have made their presentations with their assembled media, a pair of students returns to the tool to review all the class presentations. They type in comments and questions that occurred to them after their class discussion and work on their next project. Their teacher and the other students will read and respond to these at a later time. The conversation will continue.

This paper explores the potentially complementary relationship between the learning and teaching of literature and the attributes of multimedia instructional delivery systems. Our research is designed around and is driven by the assumption that the medium potentially represents a powerful means of promoting and enhancing the processes of literary understanding. Through a

ERIC

comprehensive review of the design strategies employed in both commercial and experimental multimedia applications we attempt to determine whether the design of current applications reflects state of the art theory and pedagogical approaches to the learning and teaching of literature.

This report describes the processes and outcomes of the initial stage of a three-year research and design project undertaken by the National Center for the Teaching and Learning of Literature.

Literature and Multimedia

What began as a simple merger, video with computer-aided instruction, has become a platform for instruction known as multimedia. With the capacity for orchestrated text, graphics, audio, and video, the medium appears to have great potential as a tool for complementing the teaching and learning of literature. Literature instruction currently places strong emphasis on student-centered, response-based approaches. As such, a medium that provokes, sustains and illustrates reader responses to literary text becomes a desirable tool for both students and teachers. In the role of classroom presentation system, medium for independent study, or catalyst for student collaboration, multimedia represents a tool that can aid in and accentuate these processes by providing access to and tools to work with a wide range visual and aural information. In short, the medium has the potential to serve as a tool with which teachers and students can explore and construct meaning around and their responses to literary texts.

Distinct parallels can be traced between the evolution of instructional media and recent growth in valuing student-centered explorations in the study of literature. Both design and utilization of audiovisual technology in instructional settings, and the teaching of literature, for example, were formerly based on information brokerage. Machines were viewed as efficiency tools that could mimic the mechanical process of instructional 'delivery'. Likewise, literary texts were perceived as sets of information to which the teacher permitted access.

Fundamental approaches to instructional technology have, however, seen a major shift. In recent years there has been growing recognition of the benefits students derive when actively engaged in the construction of their own knowledge. This stance provides a theoretical basis upon which a number of

ERIC

multimedia applications are currently designed (Allison & Hammond, 1989; Bransford, Sherwood, Haselbring, Kinzer & Williams, 1990; Scardamelia & Bereiter, 1991; Shapiro & Spoehr, 1992; Hannafin, 1992). Rather than deliver and drill, then, video, audio, text and graphics are currently designed to intrigue students, invite access, and prompt critical dialogue. In this role the technology simultaneously engages, delights and instructs as no other form of mediated instruction can (Iuppa & Anderson, 1988). It moreover lends itself to 'hands-on', experiential learning that invites one to relate what one knows about the world in order to move through elaborated content and construct meaning around it (Nix and Spiro, 1990).

In short, multimedia is currently viewed as a means of providing tools for exploring one's own understanding of the world: a place to hone skills autonomously (Stevens, 1987; Wilson, 1988), to experience empowerment (Snyder, 1984), to access the otherwise inaccessible (Coleman, 1992) and is also viewed as a powerful catalyst for cooperative and collaborative learning (Johnson et al, 1989). The genesis and elaboration of personal theories and the promotion of independent, critical thinking have come to take precedence over information-delivery and skills drilling paradigms of old.

The study of literature is similarly in the process of severing its roots in the information brokerage tradition. The study of literature was once, like teaching technologies, mechanical, comprehension-based and, consequently, not amenable to exploration, discovery and critical analysis on the part of students. Access to the canon was the domain of the teacher; she alone held the key to sanctioned interpretations of artistic works. Student involvement in making meaning was limited to buying into and being evaluated based on repeating information proffered by the expert. Shifts in theoretical and pedagogical groundings in the teaching of literature now, however, mirror those of instructional technology: Meaning or "understandings" of texts (Langer, 1990) are now believed to be best constructed by students, not channeled to students via an expert entity. Through socially constructed understandings, it is generally held that students will gain the critical and creative thinking skills as well as social skills crucial to academic successes (Miller's ref).

How, then, do reconstituted theories and practices in instructional technology and those in the learning and teaching of literature intersect? With both instructional paradigms currently rooted in constructionism, cooperation and socially-mediated learning, merger apparently holds great promise.

This paper treats this question by discussing methods of multimedia

ERIC

applications analysis that systematically determine the strengths and weaknesses of existing multimedia applications in order to determine whether applications possess attributes complementary to instructional paradigms that are rooted in constructionism, cooperation and socially-mediated learning. The development of a taxonomy of review criteria grounded in response-based theory is presented in tandem with the development of a database system for design attributes categorization and cross-referencing.

The Development of Literary Understandings and Multimedia Attributes

At a macro-design level, multimedia possesses the following attributes relevant to the learning and teaching of literature:

1-The technology can promote independent learning through student control of information and events (Milheim, 1988) and can thus promote student-centered, as opposed to "expert-centered" intellectual development.

2-The technology is also a powerful catalyst for cooperative learning (Palmer & Snyder, 1989; Johnson & Johnson, 1990; Webb, 1986) and as such can enhance socially mediated learning processes.

3-The technology promotes a constructionist view of learning (Papert, 1989). As such, learning takes place when students actively engage in assembling knowledge structures cooperatively, with the machine providing access to novel knowledge structures and tools with which students build.

4-Visual and aural elements of multimedia make it an engaging, visually rich learning environment and contribute to high levels of motivation and involvement (Miller, 1989).

As regards multimedia's potential capacity to enhance the literary experience at a micro-design level, a key feature is access.

-Access to the literary text itself via annotation and commentary.

-Access to multiple perspectives of a single literary work.

-Access to tools with which to build individually or group

ERIC

constructed representations of understandings developed around a piece of literature.

-Access to a dialogic space within which students, through interpersonal communication, develop understandings.

-Access to an environment and tools for personal creativity.

In its potentially supportive role in the literature classroom, the technology can be seen as complementing and enhancing the following phases for developing literary understandings as outlined by Langer (in progress).

I. Before Literary Experience

-Easing Access before Reading

As a presentation system, multimedia can provide an exquisite tool for easing entry to a literary work by providing access to supporting visual/aural information; thought-provoking images and key information that the teacher and/or group can tailor at will and utilize in consort with the front-end, discussion-based work promoted by Langer.

-Creating the Literary Experience

Multimedia offers alternative experiential modes: self-study, or the source of a social context in which literary works can be explored/experienced with others.

II. Afterward

-Inviting Understandings/Developing Interpretations

Through multimedia students can be encouraged to build meaning and develop understandings and be given aural and visual tools with which to explore, expand, clarify, and modify their understandings.

The medium can also potentially assist the student in considering multiple perspectives/interpretations by providing access to these via video, audio, graphics and text. As such the medium has the potential to invite exploration of multiple perspectives.

-Make Connections (personal, literary and cultural)

Again, students can be permitted and encouraged to do this with multimedia tools at their disposal with which they can construct as many linkages as they can support and defend.

-Sharing These, Taking a Critical Stance

The medium can serve the study of literature as a tool, a vehicle to facilitate and make more powerful the sharing of understandings through the construction of cumulative, cooperatively constructed meanings.

-Exploring the author's craft

Multimedia can supply students with a magnifying glass (among other tools) with which to examine literary works and, with the aid of multiple forms of on-line assistance, can help students make sense of a writer's artistic crafting of a piece via access to a wealth of available 'craft commentary'.

-Stocktaking

To 'leave doors open' once a piece of literature has been read and discussed, multimedia can serve as a place to return to in order to continue to probe and make sense of a work. As such it can provide the kind of independent reexamination that promotes independent as well as socially constructed envisionment building.

There has traditionally been reliance on the teacher's capacity to open doors to what was perceived as the hidden riches within a literary text. Teachers in turn have relied on texts and on students' own capacity to enter texts, to become initiates. What multimedia represents is an additional gateway; a means of access to a text's multiple dimensionality through which students, with their teacher, with peers and independently, can pass and where meaning can be built. Multimedia apparently has, then, the potential to serve as an ideal environment for exploring one's own interpretations, constructing one's own meanings and negotiating and/or defending these with peers. Because it offers student-centered experiences, it can encourage and empower independent, critical thinking.

ERIC

METHOD

In the past decade, the number of instructional applications for multimedia has mushroomed. Consequently, multimedia is currently evolving pedagogical questions which will shape the way we think about the design and the 'delivery' of instruction and, additionally, how the effectiveness of specific design approaches can be empirically researched. It is with these considerations that an iterative review and critique approach was developed.

Ten exemplary teachers of literature at the secondary level participated in the development of assessment criteria and resulting evaluation tools for multimedia applications. Their task was to specify desirable attributes of both multimedia applications and classroom contexts in which applications can play a role and contribute to this form of student-centered pedagogy. Participants reviewed both commercial and experimental multimedia applications designed for use with the teacher and learning of literature.

The evolution of our applications assessment tool will be detailed within the following developmental phases -

-Overview of the technology and its capabilities
-Exploration of the potential role of multimedia in response-based treatments of literature in the classroom
-Development of design desirata
-Drafting of assessment criteria
-Experimental application of evaluation tool
-Redrafting of tool
-Design of computerized applications database design feature taxonomy developed by practicing experts.

The outcome, a tool to determine whether applications and their use were designed to promote responsive, critical thinking and the construction of knowing, the development of understandings, and the sharing of understandings, will be presented. Records on each application reviewed for the computerized database describe and rate a particular application according to criteria iteratively established during the survey process.

OUTCOMES

This paper describes the preliminary stage of what will be a large-scale inventory and critique of existing multimedia applications developed for the teaching of literature. That is, it is a report on the processes entailed in the development of assessment tools for the critique of existing applications that is grounded in current response-based approaches. We are convinced that data compiled on application attributes and current usage will contribute to the formation of a basis for constructing a critical framework within which we can begin to make sense of the potential matches and/or mismatches of the medium and response-based views of the teaching of literature.

References

Allison, L. and Hammond, N. (1989) A learning support environment: the Hitch Hikers Guide. in McAleese, R. (ed) Hypertext: Theory into Practice. Norwoord, NJ: Ablex Publishing Corporation.

Bransford, J. D., Sherwood, R. D., Hasselbring, T. S., Kinzer, C. K. and Williams, S. M. (1990) Anchored instruction: Why we need it and how technology can help. in Nix, D & Spiro, R. (eds) Cognition, Education, and Multimedia: Exploring Ideas in High Technology. Hillsdale, NJ: Lawrence Erlbaum Associates.

Coleman, M. (1992) Teaching literature and visual arts with hypertext. Proceedings of the Conference on Instructional Technologies. State University of New York, May.

Cuban, L. (1986) Teachers and Machines: The Classroom Use of Technology since 1920. New York: Teacher's College Press.

Hannafin, M. J. (1992) Emerging technologies, ISD, and Learning Environments: Critical Perspectives. Educational Technology Research and Development, 40 (1) 49-63.

Iuppa, N. and Anderson, K. (1988) Advanced Interactive Video Design. White Plains, NY: Knowledge Industry Publications.

Johnson, R. & Johnson, D. (1986) Computer-assisted cooperative learning. Educational Technology. 26 (1) 12-18.

Johnson, R., Johnson, D. and Stanne, M. (1985) Effects of cooperative, competiti e, and individualistic goal structures on computer-assisted instruction. Journal of Educational Psychology, 7 (6) 669-677.

Langer, J. (in progress) Supporting the literary experience: The focus of instruction.

Langer, J. (1990)The process of understanding: reading for literary and informative purposes. Research in the Teaching of English, 24 (3) 229-260.

Milheim, W. (1988) Learner control options as an effective strategy. Instructional Delivery Systems, September/ October.

Miller, R. (1990) Learning benefits of interactive technologies. Multimedia and Videodisc Monitor. February.

Nix, D. and Spiro, R. (1990) Cognition, Education and Multimedia: Exploring Ideas in High Technology. Hillsdale, NJ: Lawrence Erlbaum.

Papert, S. (1986) Just a computer. Principal, 66 (2) 49-50.

Scardamelia, 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, 37-68.

Shapiro, A. M. and Spoehr, K. T. (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, April, 1992.

Snyder, T. and Palmer, J. (1986) In Search of the Most Amazing Thing. Reading, MA: Addison-Wesley.

Snyder, T. (1984) Designing computer games and simulations. address to the Harvard Graduate School of Education.

Webb, N. (1983) Microcomputer learning in small groups: Cognitive requirements and group processes. Journal of Educational Psychology. 76, 6, 1076-1088.

This research is supported under the Educational Research and Development Center Program (Grant number R117G10015) as administered by the Office of Research, Office of Educational Research and Improvement, U.S. Department of Education. The findings and opinions expressed here do not necessarily reflect the position or policies of the sponsoring agency.

ERIC

What makes multimedia systems interesting for education?

VITTORIO MIDORO
Istituto Tecnologie Didattiche, Consiglio Nazionale Ricerche, Genova, 16145, Italy

What makes multimedia systems interesting for education? Three main characteristics:

multi-channel communication;

the huge quantity of learning material which can be integrated in them;

the access facilities to the learning material.

The present understanding of learning processes must be extended if these potentialities are to be fully exploited (Kibby and Mayes, 1992). Nevertheless, studies on some aspects of the educational process have already produced useful information on how to improve the effectiveness of hypermedia in learning.

multi-channel communication

There are studies about the relationship between the type of topics in a given subject and the most suitable means to be adopted for educational communication (Sanna, 1982).

learning material integration

There are several proposals on how to integrate the learning material in a student station and these are strictly linked to the technology available. However the lack of commonly accepted standards is hindering the diffusion and development of multimedia systems, thus reducing the possibility of a wide scale testing in the educational field.

access facilities to the learning material

The limitations of free navigation are by now clear. Several studies have been carried out to combine the modes of hypermedia access with typical query languages particular to database technology. These studies, which have led to the definition of the concept of Database of Learning Material (DBLM) (Olimpo & al., 1990) have investigated the nature of multimedia documents and their pedagogical qualities, but have yet to produce any evidence of their effectiveness in the learning processes.

So far, we have considered multimedia learning environments (hypermedia and DBLM). However programming environments designed specifically for development of hypermedia systems are just as important. These tools are nearly always available to the users of a hypermedia system and the students can use them at different levels of complexity to design and implement their own hypermedia systems.

Summing up, the multimedia paradigm has made available to education new learning environments and new production tools which can be used both by professional authors and students.

In the following section we will discuss two issues:

the use of hypermedia systems as learning environments,

the use of hypermedia production environments as productivity tools within the authoring process, and as tools on which to base the development of projects in the classroom.

ERIC

Hypermedia and DBLM for learning

Two schools of thought have greatly affected the theoretical models determining the characteristics and functions of interactive systems for learning (Midoro, 1989):

Behaviourism

based on the idea of knowledge as a set of behaviour and skills to be transferred into the student's mind through a process made up of an arranged succession of stimuli and responses.

Constructivism

which considers the student as the builder of his/her own knowledge system, through interaction with a given environment.

The behaviourist model has lead to directive educational systems such as tutorials, drills & practice, etc. The constructivist model has lead to the development of the so-called "reactive" interactive systems, such as simulations, games and microworlds for learning.

"Bare" Hypermedia and DBLM cannot be classified as adaptive systems, because such systems feature an educational procedure determined by the system and a mechanism of control over the student's actions.

Nor can they be classified as reactive systems, even in hypermedia and DBLM the student has the initiative in his/her hands. In fact reactive systems imply a space for tasks and problem solving. This space appears as an environment which either changes or reacts according to the actions of the student and changes occur following a model implemented in the system.

Hypermedia and DBLMs are based on a conceptual model too; the former describes the links between the single components, while the latter shows the relations between the different entities with their respective attributes. A hypermedia system or a DBLM might be considered as a reactive system when the educational aim focuses on information retrieval skills or when the interface is isomorphic to the topics dealt with. Here the interface exploration induces, to a certain extent, topic knowledge as well. But generally speaking, multimedia systems cannot be considered as reactive.

So the kind of learning process supported by hypermedia and DBLM can be referred to neither Behaviourism which has proven efficient in the transfer, automation and evaluation of specific knowledge, nor Constructivism, which is considered effective for the acquisition of problem solving abilities.

The learning process supported by the use of this kind of systems might be considered similar to that based on the use of a book or a television programme and hence is in some way non-interactive.

In a multimedia system it is the user who decides the exploration strategy. But is it an effective learning tool?

To answer this question we tested the hypermedia system TERREMOTI with 36 students (Frau & al., 1992). The results proved that, in a hypermedia system, high quality educational communication, richness of the material available and a friendly interface are all necessary conditions, but are not sufficient to obtain a deep understanding of the contents dealt with.

To finalise the process of contents exploration in this experimentation, a task was introduced for to the students to accomplish at the end of interaction. The result was that the presence of a given task affected the choice of contents, but did not per se favour a navigation strategy aimed towards the acquisition of an interpretative model of the reality under consideration. On the whole, this experiment has highlighted the fact that optimising partial aspects of the educational process, in this case communication and access to information, does not produce per se a major improvement in learning, when diagnosis, help, feedback and motivation are lacking.

Moreover, as the student must explicitly define a navigation strategy he/she has a higher workload than that necessary when reading a book or watching a television programme.

Hypermedia systems and DBLMs require explicit and conscious reasoning about the study strategy, and this is why they are more difficult than linearly-organised material (books, TV programs, etc.). Unlike adaptive or reactive systems, the user's reasoning about the interaction strategy must here be explicit; it constitutes an indispensable condition for the learning process, which would otherwise be seriously hindered. This is why these systems have not proved particularly effective with the average student, while they are with teachers who are familiar with the subject domain investigated.

The above mentioned TERREMOTI experimentation has shown that many cognitive activities are required to fully exploit the potentialities of a hypermedia. Some of these are as follows:

- comprehension and learning of the system functionality;
- interface comprehension and learning (for access to functions);
- content comprehension and learning;
- comprehension of the type of relationship between the topics constituting the contents;
- construction of a consistent relationship system between the topics;
- definition and implementation of a learning strategy and tactics;
- definition and implementation of a navigation strategy and tactics;
- definition and implementation of methods for self assessment over learning.

A student learning through a multimedia system should be conscious of these cognitive activities and suitable tools should be provided to support his/her learning process.

Hypermedia and DBLM as authoring tool

In a hypermedia system the distinction between author and end-user is not clear cut. In fact, in many hypermedia systems, the user can easily shift from the navigation or browsing environment to the authoring environment. Moreover these authoring environments can be used at different levels of complexity, allowing even naive end-users to create personal multimedia documents. In many multimedia systems the *cut* and *paste* functions are available and through them the user can create personalised multimedia notes.

These possibilities have important repercussions in education since they foreshadow new scenarios for the learning process, such as those described below.

Personalisation of a hypermedia system

In this scenario, for example, the student personalises an existing hypermedia system by rearranging the web of relations between the various topics and the modification of the educational material related to each node. The new web might be a subset of the original net or one which is altogether different. The same also applies to the interface which could be either analogous to the original or completely different.

There are several reasons for personalising a hypermedia system. It can be a way for the student to understand thoroughly a given domain. This activity is similar to that of repeating aloud the lesson just studied in a book. Obviously, just as studying with a hypermedia system is more complicated than studying a book, the task of personalising a system also requires more complex cognitive activities, ranging from the comprehension of and the acquaintance with a subset of the system's authoring functions to the ability to rearrange the hypermedia system contents.

A further reason for personalisation might be that of demonstrating the learning of a given concept through its explanation (Pask, 1979), which in this case would consist of a hypermedia system produced by the student himself as a personalisation of an existing one.

Moreover a hypermedia system personalised by a student could be used by other students as a simplified study tool.

Co-operative production of a hypermedia system

The authoring tools available in a multimedia system allow the introduction of new relations and new nodes into a hypermedia net or new units of learning material into a DBLM. This permits the conception of systems which grow through the intervention of several users, such as where

ERIC

schools co-operate in the development of a hypermedia system for a given content domain. In this case the system grows as the activity in the schools involved in the project progresses. Work could be shared out among schools so that the material produced by one is available to all the others. This scenario is a generalisation of the approach used in the production of the well known BBC multimedia system "Doomsday" .

Classroom projects based on the use of hypermedia systems

In this scenario the project team, made up of the whole class co-ordinated by the teacher, designs and implements a hypermedia system.

An example of this approach is the project for the production of the hypertext "Visitando....Spinola", (Culotta Leccioli, 1991) developed by the fifth year class of the 'Rodari' elementary school in Genoa. In this case hypertext techniques were used to carry out activities requiring analysis, synthesis and evaluation, higher-order skills compared to those required by usual classroom practice.

In the "Visitando....Spinola" project, the pupils firstly chose a content domain for investigation which they found particularly motivating, i.e. Palazzo Spinola, a national museum in Genoa's historic centre. Then they tackled all the typical phases of a computer science project, which range from problem analysis to program testing and quality control.

At the beginning, the pupils represented the knowledge to be dealt with by the hypertext. This representation, consisting of Palazzo Spinola's structure and the arrangement of the objects it contains, was the basis of the interface.

Then they gathered information about Palazzo Spinola (grand staircase, hall, courtyards, etc.) and about the objects, which they selected and connected to each interface object.

Next they designed the system by defining three modules, corresponding to three hypercard stacks: a guide to the museum, a library and a glossary.

Finally they wrote the scripts relative to each card.

This activity was carried out during the whole school year and resulted in a product which is now actually being used in the museum. "Thus the school was connected to the local reality, both through the project contents and the final product itself: a museum guide which is really available to visitors. The students examined and used books and documents supplied both by their teachers and by other adults who co-operated with them. They worked with paper, pen, pencil and computer..." (Culotta Leccioli, 1991)

Material reuse for the production of new systems

As mentioned in the introduction, one of the characteristics of a multimedia system is the possibility of storing a huge quantity of text, still images, sound and, so far, a modest quantity of video. These systems allow authors to get educational material produced in various contexts and integrate it into a DBLM to produce new educational material. The use of a DBLM in the authoring process is described in (Midoro et al., 1992). Here we only wish to point out that a DBLM can be a useful tool in all life cycle phases of an educational system.

Indeed, the definition of courseware requirements to get a general overview of existing materials and approaches is not the only use for a DBLM. It can also be used to define specifications to detect and browse the learning material associated to each node of the knowledge structure. It can also be used in the design phase to select the specific material to be reused in the context of the new software and finally it can be used in the implementation phase, in which the material selected from the DBLM is integrated into the new courseware.

Use of hypermedia environments for courseware authoring

There are several examples of CAI programs produced using hypermedia development environments. The quality of these products is comparable to that of analogous courseware developed with authoring systems or with general purpose programming languages. However they do not take advantage of the new potentialities offered by hypermedia , which allow both directive and free navigation strategies (Siviter and Brown, 1992).

A promising approach, typical in Artificial Intelligence, is to separate aspects pertaining to the representation of subject matter from the aspects related to the learning material, which is the resource available for the comprehension of single topics.

Hypermedia also make it easier to adopt this approach for non-AI courseware as well. Separating learning material from the knowledge structure permits the use of the same learning material both for guided learning and for free navigation through the material.

An author can assign the task of knowledge structure management to the hypermedia and the task of retrieving and displaying the learning material to the DBLM .

Use of hypermedia environments for producing authoring tools

The authoring facilities of hypermedia systems can be used to develop shells for educational systems which encapsulate given educational strategies. This is due to the above-mentioned possibility of separating structural aspects of the educational process from the learning material associated to a single topic.

The HYPERDELFI system produced by the Istituto Tecnologie Didattiche is an example of this approach. This system is a new version of DELFI (Ferraris et al.1984), which assists an author in the design and implementation of diagnostic tests. In DELFI, the test production phases are:

1. Subject matter structuring (author's duty)
2. Input of such representation in the system (author's duty)
3. Production of the test item structure, showing the item number and the abilities required to perform the tasks activated by each query (computer's duty)
4. Writing of each item (author's duty)
5. Implementation of the code related to each item (author's duty)
6. Production of the test-handling strategy indicating the kind of questions to be presented and how to present them according to the student's answers (computer's duty).

In DELFI, implementing test items poses the author some difficulty, as a fairly firm grasp of Pascal programming is required.

To overcome this problem a hypertext version of DELFI, called HYPERDELFI, has been developed (Trentin & Midoro, 1992). An author implementing a test item with HYPERDELFI can exploit the interoperability of applications available in modern operating systems (Finder (Macintosh) and Windows (PCs)) and use the authoring facilities typical of a hypertext development environment (Hypercard and Toolbook).

Given the separation between the subject matter structure and the learning material, in HYPERDELFI an author can associate many isomorphic items to the same topic and let the system choose the item to be presented at run time .

Conclusions

The multimedia paradigm has made available new learning environments and new production tools for education.

This paper has discussed the characteristics of hypermedia learning environments, examining their strengths and weaknesses.

Multimedia systems can also be used in the authoring process. This offers interesting new possibilities for education, some of which have been dealt with in the paper.

As a final remark it should be noted that, in education, multimedia systems should not be conceived as new systems for doing old tasks, but as a new class of systems which open up new possibilities. Our job as researchers in education technology is to fully exploit the multimedia features to make learning processes more captivating and effective.

References

Culotta Leccioli N. (1991). Visitando Spinola: costruire un ipertesto nella scuola elementare. *Golem*, maggio, 16-21.

Ferraris M., Midoro V. and Olimpo G.(1984). Diagnostic Testing and the development of CAL remedial sequences, *Computers and Education*, 8, 407-414.

Frau E., Midoro V. and Pedemonte G.M. (1992). Do hypermedia really enhance learning: A case study on earthquake education. *ETTI*, 29(1), 299-309.

Kibby M.R. and Mayes J.T.(1992). The learner's view of hypermedia. *Learning Technology in the European Communities*. Klinver Academic Publishers, 53-65.

Midoro V. (1989). Le basi teoriche dei sistemi interattivi per la formazione. *Interazione col computer e processi formativi*, a cura di Gilli D. e Grimaldi A., Franco Angeli editore. 17-65.

Midoro V., Persico D., Trentin G. and Viarengo V. (1992). Using a Database of Learning Material in Courseware authoring. *Journal of Educational Multimedia and HyperMedia*, 1(1), 77-90.

Olimpo, G., Chioccariello, A., Midoro, V., Persico, D., Sarti, L., Tavella, M. & Trentin, G. (1990). *On the Concept of Database of Multimedia Learning Material*, Proceedings of WCCE 90, Sydney, North-Holland, Amsterdam, 431-436.

Olimpo, G., Chioccariello, A., Midoro, V., Persico, D., Sarti, L., Tavella, M. & Trentin, G. (1989). *A conceptual model for a Data Base of Learning Material*, Programme DELTA, Project D1012, Deliverable 4.

Pask, G. (1979). A Proto-Language, LP the Thoughtsticker Language. System Research, internal report.

Sanna, R. (1982). Logica della comunicazione visiva e audiovisiva. *Atti V corso nazionale per formatori*, 99-115.

Siviter D. and Brown K. (1992). "Hypercourseware", *Computers and Education*, 18 (1-3), 163-170.

Trentin G. and Midoro V. (1992). *A hypertext approach to learning assessment and student modelling*, ITD technical report.

Dexter-Groupware Model for Collaborative Authoring

ZHENG MIN
Institute of Computer Science and Technology
Peking University, Beijing, 100871, China

ROY RADA
Department of Computer Science
University of Liverpool, Liverpool L69 3BX, England

Computer Supported Collaborative Work (CSCW) is a rapidly emerging research area, which provides new opportunities and challenges to researchers, practitioners, and designers from a number of disciplines such as computer science, cognitive psychology, social psychology, and software engineering. Meanwhile, advanced network techniques provide the individual users and managers extended communication capabilities in improving both productivity and organizational management. On the other hand, the absence of conceptual frameworks which can be directly applied to the collaborative development of multimedia courseware demands a substantial amount of work to lay down the principles, guidelines, and models so that the subsequent work of integration can be smoothly and soundly pursued.

Information sharing is essential in any collaborative activity. Hypermedia offers different information presentation techniques than previous technologies, which enables people to share information. Furthermore, hypermedia provides a richer quality of educational experience. Hypermedia systems can provide us with robust objects with which to think, such as self-authoring texts that bring forth the movements of a mind focused on a particular topic [Barrett 1988]. Assessing the validity of this claim for hypermedia depends upon evaluating the learning in relation to the goals or purposes with which the learning was undertaken. Furthermore in a complex hypermedia learning environment, learning may also take place which is unrelated to specific goals but which is nevertheless valuable [McKnight et al., 1991].

As hypermedia represents information in networks rather than in lines, hypermedia presentations may fit the model of the world in peoples' minds better than traditional linear presentations. However, for various reasons, the paper form of documents still dominates the dissemination of information and massive information is available in paper form. One way to produce hypermedia effectively may be to establish a systematic connection between linear document and hypermedia and create hypermedia from traditional linear documents automatically [Furuta et al., 1990; Ritchie, 1989; Sarre et al., 1990; Nunn et al. 1988; Chignell et al., 1991; Glushko 1989].

The Dexter-Groupware Model

In order to successfully build teaching systems that people will use, we need to carefully consider the needs of those people and their current working situations. In this paper, a model called Dexter-Groupware for computer supported collaborative authoring and document reuse is presented. The Dexter-Groupware (DG) model shows the collaborative authoring and reading facilitated by international standards (SGML/HyTime) and a widely accepted hypertext reference

ERIC

model (the Dexter model). The aim of this model is to provide a conceptual framework to identify, design and develop effective facilities to support collaborative and distributed authoring and reusing of multimedia courseware.

The Dexter Hypertext Reference Model

The Dexter model is an attempt to capture, both formally and informally, the important abstractions found in a wide range of existing and future hypertext systems. The goal of the model is to provide a principled basis for comparing systems as well as for developing interchange and interoperability standards [Halasz & Schwartz, 1990]. The Dexter model divides a hypertext system into three layers, the runtime layer, the storage layer and the within-component layer (see Figure 1). The main focus of the model is on the storage layer, which models the basic node/link network structure that is the essence of hypertext.

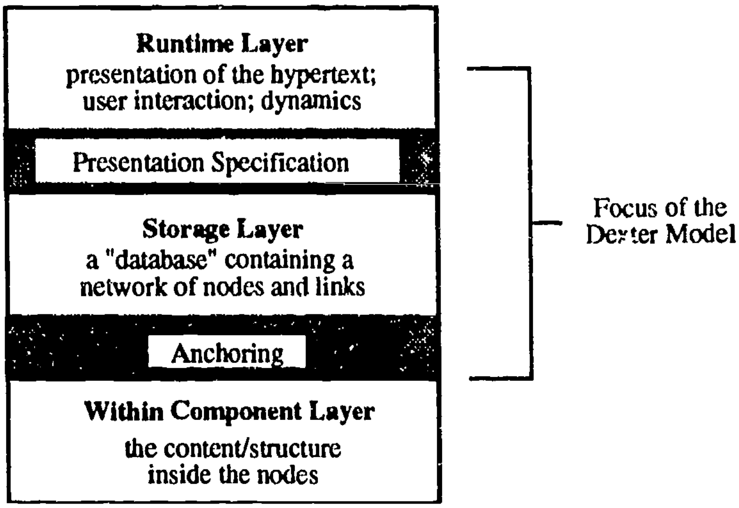


Figure 1. Layers of the Dexter model.

The storage layer only provides the mechanisms to organize components (nodes) and links to form hypertext networks without considering the contents of components, which is the task of the within-component layer. The interface between the storage layer and within-component layer is an anchoring mechanism which is used for addressing (referring to) locations or items within the content of an individual component. The runtime layer focuses on how a component/network is presented to users. Between the runtime layer and storage layer, there is a interface called presentation specifications, which provides a mechanism to encode the information about presentation into the hypertext network at the storage layer.

Dexter-Groupware Model

The Dexter-Groupware model uses the Dexter hypertext reference model as its basis. The Dexter model gives a standard information structure for hypermedia systems, but it does not include enough mechanisms to support collaborative work. The Dexter-Groupware model is an extension of the Dexter model following an open collaborative authoring architecture (Figure 2) to support collaborative authoring. In this model, the authoring space and common information space

ERIC

are established on the Dexter model with the authoring space on the runtime layer and common information space on the storage and within component layers (Figure 3).

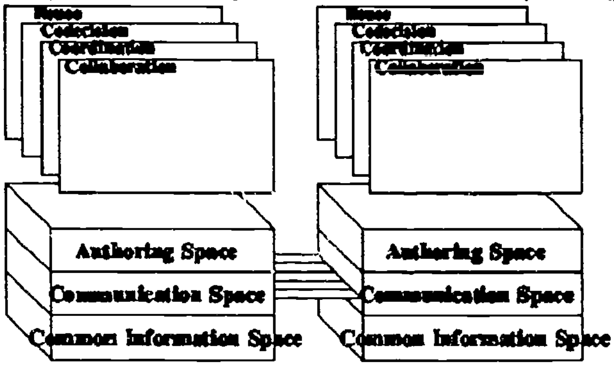


Figure 2. Infrastructure: an open collaborative authoring architecture.

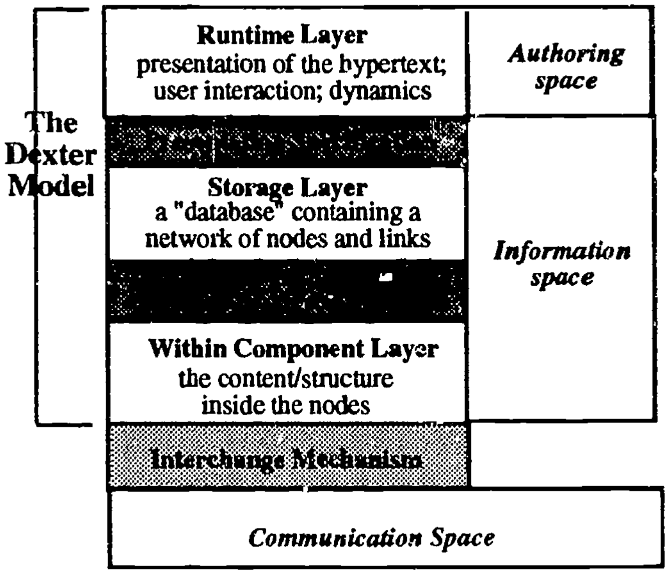


Figure 3. Dexter-Groupware Model: This shows a model for collaborative authoring based on the Dexter model.

At any time, there may exist more than one authoring space associated with the underlying hypermedia space. Each authoring space corresponds to the involvement of a particular document or project. A mapping exists between an authoring space and the hypermedia space which acts as an common information space. In the Dexter model, a hypermedia on the storage layer can have several instances on the runtime layer on which the authoring tools work. With the presentation specification of the Dexter model, a part of the information space can be mapped as a view in the authoring space and co-authors can work on the same information space through individual authoring spaces. Components in the information space can be reused at the authoring space level which is conceptually higher than the information space level.

ERIC

In the Dexter-Groupware model, the local collaboration support is reflected on the relationship between the information space and authoring space. For remote collaboration, the collaborators in a project may work on more than one information spaces at different locations. This will require information interchange among different information spaces, which is the main task of the interchange mechanism in this model.

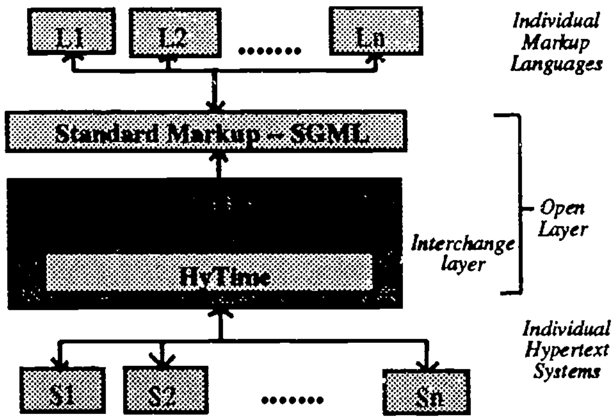


Figure 4. SHyD Model: this figure shows the basic concept of text and hypermedia conversion and interchange through standard representation.

This interchange mechanism of the DG model is based on the SHyD model [Zheng & Rada, 1993] show on Figure 4, which is dedicated to bi-directional conversions between text and hypermedia and information interchange for both text and hypermedia. The basic principle of the SHyD is using international standards (SGML/HyTime) upon a widely accepted model (the Dexter model) as intermediate representations to support information interchange.

MUCH Collaborative Authoring Environment

A prototype of the collaborative authoring environment based on the Dexter-Groupware model has been developed under the name of MUCH (Many Using and Creating Hypermedia). This environment includes a collaborative authoring system -- the MUCH system and an import/export system associated with the MUCH system to support information interchange between MUCH and other text and hypermedia systems.

MUCH Collaborative Authoring System

The MUCH system [Rada et al., 1991] is modeled after the Dexter model. Its components are the typical interface, logical model, storage components. The authoring space of the MUCH system is developed on the Andrew multimedia windowing environment [Grantham et al. 1987] and the logical model and storage components are implemented in a B-tree database. One of the principles of the interface design is that simplicity is a virtue. Users should not need to know or do more than they want [Carroll, 1985]. Based on the positive experiences with the KMS split-window approach [Akscyn et al., 1988], we have also presented the hypermedia information on a split screen, with the fold-unfold outline in one part and content of a selected node in another part. When the user selects a node in the fold-unfold outline, the associated blob appears in the blob window.

While the logical model contains a graph, the user sees at any time a hierarchical view of this net. In MUCH, the fish eye view feature is implemented as the "folding/unfolding" of thesaurus terms, indices and outlines of documents. Different headings can be distinguished by labels of different link types. Annotation and discussion are also supported by a 'typed link' as a communication mechanism among co-authors. The MUCH system supports automatic traversals of the graph. These traversals may be likened to question answering processes in a deductive database [Niemi & Jarvelin, 1992].

Via the link types, users characterize their contribution. If the user creates a node and links it to another node with the link type 'comment', then an annotation has been made. In the generation of views of the information in the system, the user can request to traverse only 'comment' links. A thesaurus is just another type of semantic net, and support for thesauri is provided in the MUCH system by the establishment of link types for thesauri. One link, called 'document-pointer', goes from a thesaurus term to a node that corresponds to part of a document. Again one can filter the hypermedia by choosing to select or avoid thesaurus links.

The MUCH Hypertext Model is a simplified Dexter model. It adopts the link mechanism of the Dexter model. This model focuses on link components. In the MUCH database, there are three kinds of components -- link, node-info and blob. The link component connects exactly two nodes and also contains some attributes of the link, such as author, date, and link type. Node-info is the information about individual nodes. It includes name, type, class, author, date, updaters, index terms and others. A blob includes the content of a node. Some other information like selected-credit, inclu .ng blob-selected-credit and user-selected-credit, is also included in the database.

SGML-MUCH (SM) Import/Export System

To facilitate reuse of information and the integration of MUCH into the wider range of collaborative work flow, tools have been provided for importing and exporting documents. The SM system is a prototype system for hypermedia interchange and conversion. In particular it is used to interchange information between the MUCH system and other systems and to receive information from both linear and hypermedia forms.

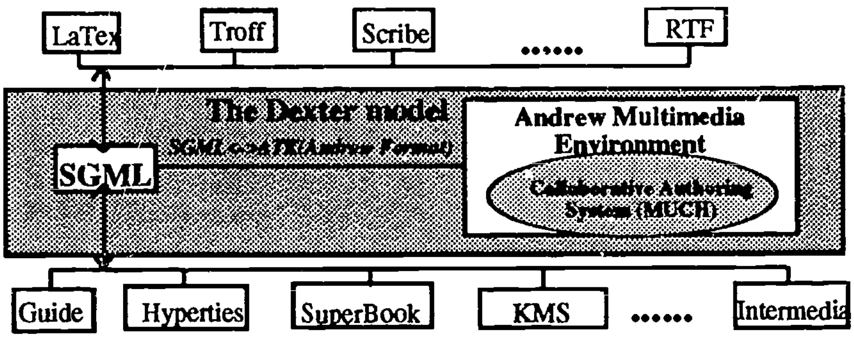


Figure 5. The structure of the SM system.

The SM system has the capability of exporting a part of the MUCH database into a linear document with all hypermedia information. This exported linear document can be edited and processed by conventional document processing systems while retaining all the information. On

ERIC

the other hand, because all the links and authoring information like author, date and updaters are included in the exported document, this document can be further used to interchange information with other systems, both homogeneous (MUCH <-> MUCH) and heterogeneous (MUCH <-> others).

The structure of the SM system is shown on Figure 5. The SM system is based on a simplified SHyD model, where SGML plays a central role but not HyTime because of some practical restrictions. In the SM system, a SGML DTD (Data Type Definition) which can represent both text and hypermedia structures is defined following the Dexter model and is used as an intermediate format. With this mechanism, the MUCH system can accept information from either text document or other hypermedia systems and process the information by taking advantage of the capabilities of the MUCH, such as reuse and outline generation. The information processed by MUCH can be exported to a SGML document which can be then translated into other text or hypermedia systems.

Experiences with Using the MUCH for Teaching

The MUCH system was used in teaching a class on hypermedia at the University of Liverpool. Students met in a computer laboratory and each student in the group had access to a workstation connected to the MUCH system. On the network the book *Hypertext: from Text to Expertext* had been installed. having been actually written using an earlier version of MUCH. During classes the students and professor would study and discuss the contents of the book (which, amongst other things, looked at the manipulation of text and the enhancement of communication). The collaborative hypermedia system used, actually illustrated (and so re-emphasized) the points made in the book. Thus the MUCH system provided both the theory and the practice of a hypermedia environment.

The students were required to do exercises in groups using the MUCH system and so, again, the theory being learnt was constantly reinforced. Two exercises in particular merit description here: one involved an assessment of MUCH-supported collaborative authoring and the other involved automatic peer evaluation. In the first exercise the students were asked to create in small teams a short essay about the pros and cons of collaborative authoring with the MUCH system. The most frequently reported advantages of MUCH usage were its support of collaboration and the ease with which annotations and other manipulations could be performed While MUCH is partially synchronous, its failure to be truly synchronous engendered multiple complaints. The familiarity of paper versus the unfamiliarity of MUCH was also noted as a disadvantage for MUCH.

The second assignment of the class was for students to make, at least two, comments on the manual for the MUCH system which had been loaded into dthe MUCH system. They were told that their grade for this exercise would be based on how other students responded to their work. More specifically, this response from other students would be measured in two ways:

Selection credit: how often the text blocks which the student wrote or modified were 'read' by other students. 'Read' meant 'selected' because the computer kept track of every selection but did not know whether or not information was actually 'read'. A student did not get any credit for selecting his own text block.

Creation credit: how often the nodes which the student wrote became the ancestor for nodes created by other students. For example, if a student created a node and another student attached a comment to it, then the first student received a 'create' credit.

The results of the exercise were quite interesting. A number of phenomena that one sees in the advertising world were employed by students to get more credit. For instance, some students created flashy node names and node contents to get attention (e.g. "Bethan's mega exciting comment!"). Other students tried to place their nodes near the top of the outline so that other students were more likely to see them.Third, students who made minor amendments to other nodes which had high importance managed to collect large amounts of credit.

Students did truly enjoy the discussion about the results of the exercise and made further observations about computer-supported peer evaluation. The students felt that they should be more actively involved in the assessment process. They proposed that the system should allow the students to give grades (A-F) to a selected node and this information would be stored in special ways. The questionnaire would allow the student to evaluate others along multiple dimensions. The questionnaire should not be lengthy, ambiguous, or difficult to operate. However, evaluating each node could be tedious and discourage users of the system. Alternatively, evaluation could occur at the end of the work.

Conclusion

The Dexter-Groupware (DG) model, along with the SGML-HyTime-Dexter (SHyD) model provides a framework for collaborative hypermedia. The DG model incorporates an interchange mechanism into a standard hypertext reference model -- the Dexter model -- to enhance the communication capabilities of the Dexter model. The interchange mechanism, namely the SHyD model, is dedicated to information interchange and particularly to bi-directional conversion between text and hypermedia. This interchange is achieved by employing several international standards and by using the DG model to support collaborative work and document reuse.

A collaborative authoring and reuse environment has been developed under the name of Many Using and Creating Hypermedia (MUCH) based on the the DG model. Our experiences in collaborative authoring and publishing with MUCH have shown that such tools can effectively improve the production of large volumes of multimedia.

This paper mainly focuses on a model for sharing collaboratively developed multimedia material. Collaboration and reuse may improve productivities within a well-coordinated organization and increase the usability of resources. Improving product reusabilities and methods of collaboration will directly impact the creation and dissemination of hypermedia information.

Acknowledgment:

This work would not have been possible without the support of the Commission of the European Communities under DELTA Project D-2006 OSCAR (an Open System for Collaborative Authoring and Reuse) nor without the help of our colleagues on the OSCAR team in Liverpool, particularly Martin Beer, Weigang Wang, Antonios Michailidis, Damon Chaplin and Chaomei Chen.

References

Akscyn, R., McCracken, D., and Yoder, E. (1988), KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations, *Communications of the Association of Computing Machinery,* 31(7), 820-835.

Barrett, E. (1988), Hypertext and Teaching of Writing, *Text, ConText, and Hypertext*, MIT Press, Campridge, Massachusetts, 111-129.

Carroll, J. M. (1985), Minimalist Design for Active Users, *Human-Computer Interaction - Interact '84*, North-Holland, Amsterdam, 39-44.

Chignell, M. H., Nordhausen, B., Valdez, J. F. and Waterworth, J. A. (1991), The HEFTI Model of Text to Hypertext Conversion, *Hypermedia, 3(3)*, 187-205.

Furuta, R., Plaisant, C. and Shneiderman, B. (1990), Automatically Transforming Regularly-Structured Linear Documents into Hypertext, *Electronic Publishing, 2(4)*, 211-229.

Glushko, R. K. (1989), Transforming Text into Hypertext for a Compact Disk Encyclopedia, *Proceedings of CHI'89*, Association of Computing Machinery, New York, 293-298.

Grantham, D., Robertson, J., Subasic, K., and Langston, D. (1987), *A Guide to Andrew*, Information Technology Center, Carnegie Mellon University, Pittsburg.

Halasz, F. and Schwartz, M. (1990), The Dexter Hypertext Reference Model, *Proceedings of the Hypertext Standardization Workshop (National Institute of Standards and Technology Special Publication 500-178)*, U.S. Goverment Printing Office, Washington DC. 95-134.

McKnight, C., Dillon, A. and Richardson, J. (1991), *Hypertext context*, Cambridge University Press, Cambridge UK.

Niemi, T. and Jarvelin, K. (1992), Advanced Query Formulation in Deductive Databases, *Information Processing & Management, 28(2)*, 181-200.

Nunn, D., Leggett, J., Boyle, C., and Hicks, D. (1988), The REXX project: A case study of automatic hypertext construction, *Technical report TAMU 88-021*, Hypertext Research Lab., Texas A&M University, College Station, Texas.

Rada, R., Zeb, A., You, G. N., Michailidis, A., and Mhashi, M. (1991) Collaborative Hypertext and the MUCH System, *Journal of Information Science, 17*, 191-196.

Ritchie, I. (1989), HYPERTEXT -- Moving Towards Large Volume, *The Computer Journal, 23(6)*, 516-532.

Sarre, F., Seidt, M., and Guntzer, U. (1990), HyperTex -- A System for the Automatic Generation of Hypertext Textbooks from Linear Texts, *Proceedings of Database and Expert Systems Applications Database and Expert Systems Applications*,ed. K. Wagner, Springer-Verlag, 62-68.

Zheng, M. and Rada, R. (1993), SHyD -- A Model for Bridging Text and Hypermedia, *1993 ACM Computer Science Conference*, Indianapolis, Indiana.

Sailing Perseus: Instructional Strategies for Hypermedia in the Classics

Kenneth Morrell, St. Olaf College
Gary Marchionini & Delia Neuman, University of Maryland at College Park

In the world of ancient Greece, sailors had two options when attempting to go by sea from one point to another: they could sail directly in the open sea, navigating by the sun and stars, or they could follow the coast. The choice depended on the destination, the time constraints (if any) for reaching the destination, and the experience and confidence of the sailor. Either method might be appropriate, and both had their risks. If we compare using Perseus to sailing in the Mediterranean, we find that many of the issues remain the same. How users approach the materials will reflects their ultimate goal, the time they have to reach the goal, and their level of expertise with respect to both the content and the medium. In the case of Perseus, however, the shape of the sea and the distances between "ports" keep changing. Furthermore, the navigator can rely on no long tradition of knowledge about the "sea of information" to make "sailing" any easier. Our experience in developing and evaluating Perseus over the past four years has illustrated that instructors and students have used distinctly different sailing techniques and strategies. These early experiences point the way for other sailors to follow and provide guidance to those who would sail in other environments or build lighthouses and instruments for other learners.

Perseus 1.0 currently contains approximately 6000 images and over a hundred megabytes of textual information and indices. In its most basic configuration, Perseus is a HyperCard-based system that comes on CD-ROM with texts and digital images in PICT format. The images also come in an analogue format on a companion videodisc, which also includes approximately eighteen minutes of motion video in the form of four "Guided Tours" of Olympia, Isthmia, Eleusis, and Delphi. The textual and visual information provided on the CD-ROM includes primary texts in Greek and English, an historical overview, an atlas, and a variety of navigational tools. Perseus is more a library and museum than a textbook and thus provides opportunities for students to work at various levels (Crane, 1989; Mylonas, 1992).

Several examples of computer-based materials have shaped ideas about how computer technology can influence instruction in the humanities and specifically in classics: Met, a program that tutors students in the scansion of Homeric poetry; HyperLatin, a package of texts with notes and vocabulary and a field for students to write their own translations or notes; HyperMyth, a set of short entries on mythological figures and events; HyperGreek, an application that drills students on the morphology and grammar of classical and Biblical Greek; Aristotle's Greek Tragedy Construction Kit, a hypertextual examination of Aristotle's Poetics; and the Latin selections in the Transparent Language series, which offer students grammatical and lexicographic information in addition to the Latin text. Unlike these materials, Perseus is so rich and complex that it simply does not fit conventional notions of applications with preconceived pedagogical objectives. Beyond the capability to connect a set of annotated nodes in the database, however, Perseus confronts the users with an enormous amount of information and, in its infancy, provides few preconceived ways of using it in the classroom.

This design reflects the philosophy of the developers: "If you build it, he will come." And if "he" or they come, they will figure out what to do. Perseus was designed as an "open system" to allow maximum freedom of use. This design philosophy is reflected by the dominance of implicit rather than explicit hypertexual links (Mylonas & Heath, 1990). At any point, users can traverse to specific Perseus nodes in multiple ways (e.g., through menus, typing destinations, or using HyperCard commands). This open design provides the Perseus sailor with multiple options at any time--freedom that requires navigational decision making and skill. Naturally, one can use Perseus as one would use any set of primary texts, explanatory materials, and images. However, the amount of available information and the medium within which it exists challenge instructional paradigms that were dev oed in the world of print-based materials. There has been considerable ambival among both faculty and students as they struggle to identify the best way to "sail" the unknown Perseus sea.

In the following paper, we will discuss examples of Perseus use drawn from four college-level courses taught during the 1991-92 academic year. In discussing each of these case studies, we will describe the assignments instructors used to integrate Perseus in their courses and summarize the results of our attempts to evaluate Perseus as a teaching and learning environment.

Case Study 1

At Ball State University in the spring semester of 1992, Professors Martha Payne and Walter Moskalew integrated Perseus into Classics 105: Classical Culture, a general studies course on Greek and Roman civilization. For their sections of approximately twenty-five students, the professors each demonstrated Perseus in the classroom, prepared instructional materials, and created a structured assignment. Both instructors used the "path" function to design their assignments. A "path" is a set of predetermined locations in the database that students can access sequentially. Payne developed an assignment that called for students to follow a path that began with a passage from the historical overview and then took students through selections from the texts of Herodotus and Pausanias. Other stops on the path included a representation of Pan from a bell krater; the atlas with the locations of Sparta, Eretria, Marathon, Platea, Athens, and Delphi plotted; a satellite photograph of the Peloponnese; topographical images of Marathon; and site plans for Athens and Delphi. For nearly all of the twenty-four stops, or "footprints," on the path, students were asked to respond by answering specific questions on an accompanying worksheet or by printing site plans and modifying them. The objective of the assignment was to expose students to various sources of information about the First Persian War in order to have them develop an understanding of the events and their significance by relying on primary sources rather than on popular misconceptions about the Battle of Marathon.

The objective of Moskalew's assignment was to familiarize students with the development of vase painting from the proto-geometric period (11th c. B.C.E.) to the period of Attic black figure vases (late 6th c. B.C.E.). Students followed a path that included stops at twenty vases. They were to look for specific decorative patterns and design characteristics and keep track of them on an accompanying worksheet. They were also to print (from the database) a summary card for one vase identified on the worksheet.

Students reported spending between 4 and 8 total hours in an average of 3.7 study periods to complete Payne's assignment and between 3 and 3.5 hours in an average of 2.7 periods to finish Moskalew's. The students expressed overall satisfaction with the assignments for two reasons. First, the assignments were well-focused, task-oriented,

ERIC

and accompanied by detailed directions. Second, the assignments differed from other text-based interpretive projects by making extensive use of visual images--an approach that gave the students a greater sense of directly experiencing the culture of the ancient Greeks. Pictures seemed far more immediate and compelling than texts as primary materials. Part of their overall satisfaction might have stemmed from the nature of the assignments themselves-- task-oriented, specific, and limited in scope. While the assignments may not have motivated students to explore the database and make independent discoveries, the tightly structured sequence of tasks provided them with an immediate measure of success.

Students' negative comments clustered around three issues: the speed of the computing environment, especially with regard to accessing images over the network; the number and availability of workstations; and the possibility of becoming disoriented in the database. Although the first two problems are functions of the general computing infrastructure, they did play a role in the third. Certainly the scope and diversity of information in the database and the multiple means of moving through the materials could pose a problem for students who find themselves outside the narrow and clearly defined limits of specific assignments, and there were instances when students had to quit and simply begin again because they felt lost. However, the risk of disorientation was at least partially the result of the computing infrastructure and students' unfamiliarity with the Macintosh environment. First, the slowness of the network often gave students the impression that the system was not functioning properly; second, the limited accessibility of the materials prevented students from spending enough time with the system to become thoroughly familiar with its operation and to gain much confidence in their ability to explore Perseus with greater efficiently and productively.

Collateral Support. The experiences of the instructors and students with Perseus at Ball State illustrate the importance of collateral support in the process of integrating computer-based materials into the curriculum. Collateral support comprises three elements: a supportive network of users that will sustain the continued development and refinement of courseware, a student population that is steadily improving in their ability to use electronic information, and a growing number of faculty members in other disciplines who use computer-based materials and allow students to refine their strategies for working with multiple and varied sources of information. The process of integrating technology begins with an instructor's achieving facility with the computing infrastructure and becoming thoroughly familiar with the materials. The use of Perseus proved successful at Ball State because the instructors devoted enormous time and effort to exploring Perseus and designing applications that were well-suited to the requirements and objectives of their courses. Moskalew estimated that he spent as much as twenty-five hours just creating the one assignment on vase-painting. Not all faculty members are as proficient with workstations and electronic resources as Payne and Moskalew; many do not share the same commitment to technology. Furthermore, even proficient and committed instructors cannot sustain that level of intensive activity evidenced at Ball State over a period of years.

The second element of collateral support is the general level of the students' computing and hypermedia experience and their ability to work effectively with complex databases. To raise the overall chances for successfully integrating electronic materials as complex as Perseus, instructors should expect their students to expand their familiarity with computing beyond simple word processing. Even encouraging students to use campus networks introduces them to a wider range of possibilities and prepares them for working with electronic sources of information. This element of collateral support as well as the other two depends on the development of a computing infrastructure that ensures adequate access to information for students in their study environments, for faculty members in their offices or studies, and for both students and faculty members in classrooms.

The final component of collateral support comes from other faculty members who work to incorporate computer-based materials into their curricula. Over the course of a student's experience in school, s/he develops a set of strategies for processing information presented in lectures, discussions, and printed texts. Students will become more adept and confident in using computer-based information as lectures become more interactive and incorporate more than one simultaneous source of information; as discussions increasingly address computer-based materials and even, in some cases, include discussion that is electronically mediated; and as assignments for individual study encompass a more varied assortment of resources ranging from conventional books to multimedia applications. The technology itself will become more transparent, and students will begin to focus on the quality and relevance of the information, not its mode of delivery.

Case Study 2

In the spring semester of 1992 at the University of Maryland at College Park (UMCP), Professor Eva Stehle incorporated Perseus into Classics 330: Greek and Roman Religion. The twenty-eight members of the class were able to use Perseus on three dedicated workstations in the library. Students were expected to work in three-person teams in the preparation of oral reports. The assignment called for them to find images of an assigned god, the temples associated with his or her worship, and information about the religious practices related to each deity. Students were to collect this information and organize it into a path, which would serve as the mode of presentation for a fifteen-minute class report. In contrast with the assignments designed by Payne and Moskalew at Ball State, Stehle's assignment was considerably more open-ended. The depth and quality of the students' reports varied so widely that Stehle ultimately decided not to give letter grades but to judge the projects as either satisfactory or unsatisfactory.

Observation of various groups suggested that the most successful teams based their work around an integrative, conceptually-based thesis, while the less successful teams followed a strategy driven simply by their ability to access information in Perseus. The capability to access and collect information rapidly did not ensure a clear and cogent presentation. As Allison Mulligan, one of the members of the evaluation team, noted: "Perseus is a tool that allows students to be intellectually lazy if they choose to be. To use it well requires initiative and effort. If the effort is missing, the results will be minimal."

Perseus played a far greater role in Classics 330 than it did in Classics 105 at Ball State. Perseus represented one of the primary sources of information, provided the framework and presentation medium for the students' reports, and served as a means of reviewing for the first examination and completing the final take-home exam. A majority of students reported using Perseus between 10 and 14 times. The experiences of both the students and the instructor were mixed. The four most positive aspects of

Perseus for the students were the wide range of information, the ease of accessing information, the potential to integrate different types of information, and the quality and number of images. There were, however, corresponding problems. The broad range of the materials also meant that depth in some areas and basic coverage in others was lacking. It was in this area that a divergence between the expectations of the students and those of the instructor grew more pronounced. While the students appreciated the finite scope of the information, especially when preparing reports and reviewing for examinations, the instructor expressed considerable frustration over the inadequacies of the materials.

While a large majority (23) of the students found Perseus "useful" or "very useful" in helping them find information for their reports, students were less sure of their ability to use the system effectively. When responding to the question, "Rate how confident you feel about finding information in Perseus (assuming that the information is actually in the system)," almost half the students (13) said that they were "confident" or "very confident" six were "neutral," and 8 reported that they were "unconfident" or "very unconfident." The variation between the perceived usefulness of Perseus and the students' confidence in using the system also suggests something of a paradox: when faced with the alternative of the library as a source of information, the students responded positively to Perseus; but when it came to using Perseus to locate information, they were less enthusiastic about their chances for success. Their relative lack of confidence might also reflect the difference between the idea of simply finding information within a limited system and getting at the most important or crucial elements in order to form a well-documented assessment. As the instructor noted in the evaluation of the students' first reports, students' having greater access to information did not necessarily lead to their gaining a more clear, accurate, and synthetic understanding.

Students responded most positively to the quality of the images and the usefulness of Perseus in helping them to find thematic connections among different types of information. The juxtaposition of a large textual database with a sizable collection of images that goes well beyond the scope of any single printed source certainly influenced the students' perceptions. But again, the students' comparative viewpoint contrasted with the instructor's, who was troubled with the lack of explicit links among the sources of information. For the instructor (who, on the basis of her experience with the sources and scholarly background, sees connections between images and texts that are not explicitly noted in the system) the impression is less satisfying than it is for students (who in many cases are experiencing their first exposure to the primary sources themselves). Both the students and instructor agreed that the use of images from Perseus added a sense of "three-dimensionality" to the lectures.

Shaping Expectations. The success of Perseus in Stehle's Classics: 330 was constrained by the expectations of the instructor and the students and underscores the importance of users having a clear understanding of the limits of the materials. The physical characteristics of printed materials allow the users to gain a fairly accurate sense of the limitations of a resource before they start reading sequentially or begin a careful search for particular items of information. Users have a far more difficult time gauging the suitability of computer-based materials because many of their strategies for surveying the scope or complexity of a printed resource simply do not transfer to the electronic medium. Developers of large databases face the challenge of making massive amounts of information manageable and accessible in a manner that belies neither the richness of the resource nor the gaps. As these databases become increasingly complete, the gaps may become far more important and manageable than the contents.

Case Study 3

At Bowdoin College in the fall of 1991, Neel Smith used Perseus as the basis for a set of assignments in his course, Archaeology 101. Seventy-five students participated in the course, which serves as the introductory course for archaeology and is required for students who major in classics or archaeology. Three of the four assignments for the semester involved Perseus, which was available to students through any Macintosh connected to the campus network. The first assignment called for students to use a set of Landsat images to determine whether the traditional assumptions linking environmental factors to the location of Neolithic and Early Helladic communities were valid. The second asked students to explore textual and visual information to ascertain whether the Parthenon as a cultural artifact expressed Pericles' singular vision of the Athenian ethos or whether other influences were at work. Finally, as a take-home section of the final examination students were to design a monument at Olympia to commemorate the victory of Diagoras--basing their ideas for the work on Pindar's Seventh Olympian Ode, Pausanias' description of the statues at Olympia, and the inventory of archaeological and topographical images.

One of the most significant aspects of Smith's use of Perseus was the emphasis he placed on the students' working as members of teams. Students were free to form groups generally consisting of three members. Members of each group were free to segment the responsibilities according to each individual's abilities or proclivities. Smith supported this arrangement of self-directed participation by evaluating the reports with a system of checkmarks instead of letter grades and by having the assignments account for only twenty-five percent of the students' final grades. Consequently, students felt less pressure to make sure that each member contributed "equally" to the group's work by performing all the tasks for each assignment.

Collaborative Assignments. Assignments that allow students to work in groups appear to work extremely well to support examinationn of a wide spectrum of primary resources, because the individual members of the group can turn to each other for help in overcoming technical problems and difficulties. Students were, in fact, convinced that the group approach enhanced the depth of their learning. From Smith's perspective, the assignment on the Parthenon was less successful than the assignment on Diagoras' monument because the students were simply overwhelmed with relevant information. Both Smith and the students themselves realized that they tended to rely more on verbal than on visual evidence when confronted with the complicated task of evaluating and integrating a large number of visual images and written sources. For the assignment of Diagoras' monument, Smith responded to the students' frustration by providing two paths: one that dealt succinctly with the project and presented a limited number of sources and an optional second path with more information. Clearly, success in designing assignments will require developers to keep the range and amount of information in line with the students' ability to evaluate their discoveries and formulate clear ideas based on them.

One should also note that the dynamics of group assignments can run counter to the individually-centered expectations of the educational system. As students address the higher-level objectives of developing ideas and formulating arguments, members in a group must assume roles that go beyond helping one another in locating information and must evaluate their peers' perspectives and modes of expression. Some students expressed displeasure with the "inefficiency" in the design of group assignments: rather than proceeding directly to their individual conclusions, students had to consider alternatives posed by group members and to muster evidence in support of personal ideas and ways of expressing them.

Case Study 4

In the spring semester of 1992, Gregory Crane used Perseus in Literature and Arts A52: Classical Greek Literature and Fifth Century Athens, a course in the core curriculum at Harvard. Perseus played a central role in illustrating lectures and in the meetings of the discussion groups, and students were expected to complete two assignments using Perseus in a laboratory with networked Macintoshes and supported by a Perseus teaching assistant. The first assignment focused on the topic of authority and legitimacy in Aeschylus. After having read a number of Aeschylus' plays, students used the English-Greek word list in Perseus (a feature that allows users to search the definitions of Greek words in the lexicon for specific terms) to find Greek words that contained "king" in their definitions. Students then checked the definition of each applicable word to identify the most important words and looked to see which words appeared most frequently in Aeschylus. The assignment called for students to determine what Aeschylus' the primary word for "king" and to see if variations of this term also appeared as words other than nouns. Students performed the same analysis for the stem "tyran" with the ultimate objective of identifying the primary expressions for "tyrant" in Aeschylus. Students took this analysis one step further and used the "Show Usage" feature to locate each example of how Aeschylus used the terms. By looking at the English translations of each passage where the term was found, students determined whether Aeschylus blurred the distinction between the terms for "king" and "tyrant." As the final component of this assignment, students prepared a path to document their conclusions about how the distinction between the two terms influenced their interpretation of Aeschylus' Oresteia.

For the second assignment, students addressed two questions: "How did the Greeks represent themselves in art?" and "How did the Greeks talk about themselves?" For the first part of the assignment students chose one aspect of Greek society as documented in an exhibition at the Fogg Museum entitled "The Social Context of Greek Art." Students found objects that were in the Perseus database but not included in the exhibition to provide additional support for their observations. For the second part of the assignment, students used the skills they developed in the first assignment to search for expressions in the textual database that would illustrate how the Greeks viewed themselves. Their observations took the form of a paper approximately five pages in length.

Privileging Views of Students. According to the questionnaires, students used Perseus between 5 and 30 times, with a mean of 16.2 sessions. The results from interviews show that students spent between 4 and 10 hours using Perseus to complete their assignments. The reaction of students to the assignments was mixed. They believed that Perseus enabled them to learn "differently." Several reported in interviews that they had made personal scholarly discoveries. For example, one student noted how he explored an observation he made about fully-armed Hoplite soldiers' not wearing sandals on vase paintings but having sandals in drawings found in printed secondary sources. Since this student--like many others in the course had never studied ancient Greek before this experience--such a discovery is remarkable.

Other students, however, complained about the attention to detail required by the assignments and argued for a broader, more thematic approach. They said they would prefer to read an established scholar's analysis of Aeschylus' terminology rather than trying to discover it for themselves. Their comments did not reflect avoidance or laziness but their concerns about learning the best interpretations available. The access through Perseus to a sizable amount of primary sources of all varieties permits instructors to design assignments that privilege the perspectives of students over the observations of specialists, who in the past were the only ones who had similar opportunities to study the primary material. Some students, of course, will find this situation uncomfortable. Those who approach a course such as Literature and Arts A52 with the expectation of developing and practicing basic research skills will view Perseus-based assignments differently than those who expect assignments that will call for them to survey prevailing scholarly opinions. The findings from Crane's course underscore the fact that students come to courses with differing perspectives, just as instructors address the same topic with differing approaches. If Perseus is to change the nature of teaching and learning, helping students understand how research-based learning differs from the more traditional lecture approach must be a critical consideration. Making students aware that the personal construction of knowledge is dependent on close scrutiny of primary materials may be a useful first step.

Summary

Each of the instructors whose assignments form the basis of this discussion approached Perseus with the idea of creating ways for students to study aspects of the ancient Greek world in ways that would have been cumbersome at best with conventional sources. Even those assignments that tended to focus on a single type of information (for example, Moskalew's assignment on the conventions of vase painting and Crane's assignment on the use of certain words in Aeschylus) would have been nearly impossible without Perseus. It is doubtful that any casual user would have approached the materials in the same ways with comparable success without the assignments to guide their study.

In every case study, at least one assignment called for students to form conclusions about ancient Greece based on a variety of dissimilar sources. While the elements in the database may exist as finite, discrete elements, the instructors viewed them as parts of a synthetic whole, an interconnected set of a society's artifacts, and attempted to convey that sense to their students. Each of the case studies illustrates a particular issue that has implications for instructors who intend to integrate complex multimedia materials like Perseus into their curricula. Such resources enable instructors and students to engage in self-directed, synthetic learning that is driven by access to primary material--that is, to sail unknown regions according to their own levels of comfort and confidence. Sailors embarking on the new seas of information, whether they choose to head out over open water or follow the coast, might keep the following matters in mind. First, instructors need collateral support in the form of an adequate computational infrastructure, which includes both the material resources associated with computing (computers, networks, and data) and appropriate intellectual support (a network of informed, experienced users). Second, expectations matter! This is especially true in open systems like Perseus that allow instructors and students great

ERIC

freedom of access. Although rigid preconceptions about the role and importance of technology are problematic, instructors must be clear about how much direction to incorporate in their assignments. Generally, time investments are heavy "up front" before students use the system for highly directed assignments (e.g., creating paths to follow). Investments in time are heaviest while student use of the system for exploratory, self-directed assignments since students. Third, hypermedia-based materials by their nature challenge modes of instruction, study, and evaluation that are grounded in a competitive, individualistic educational system. Finally, technology motivates instructors and students in the humanities to privilege the perspectives of students gained through the application of basic modes of research. Providing students opportunities to work as scholars privileges them on the one hand and demands new learning strategies and perspectives of how their time is used on the other.

References

Crane, G. (1988). Redefining the book: Some preliminary problems. Academic Computing, 2(5), p. 6-11, 36-41.

Mylonas, E. (1992). An interface to classical Greek civilization. Journal of the American Society for Information Science, 43(2), 192-201.

Mylonas, E. and Heath, H. (1990). Hypertext from the data point of view. Paths and links in the Perseus Project. In Hypertexts: Concepts, systems, and applications. Paris: Cambridge U. Press.

A Prototype Pen-input Mathematical Formula Editor

YASUTOMO NAKAYAMA
Tokyo Research Laboratory, IBM Japan
5-19, Sanban-cho, Chiyoda-ku, Tokyo 102, Japan

Introduction

Formula Editors with Keyboard and Mouse

In most existing CAI applications and word processors, it is difficult to input mathematical formulas. We developed the Mathematical Formula Editor (MFE) [Nak89] to remedy this situation, and integrated it into our CAI applications [Aka92, Hid92]. Recently, some word processors on personal computers have also been equipped with an equation edit‹r or formula typesetter. These formula editors allow a user to input formulas with a keyboa and mouse. Op‹‹ations are relatively easy because most of the editors are designed in accordance with th "What You See Is What You Get" (WYSIWYG) concept. It is practical to use a keyboard and mouse because these devices are very popular at present.

Formula Editors with Pen Interface

Mathematical notations have a two-dimensional extent, so that input of mathematical formulas is similar to that of graphic objects. In this respect, a mouse is useful. But mathematical formulas include characters, and in this respect a pen interface has the merits of both a mouse and a keyboard.

Many devices and software systems for pen-input have been developed in recent years. We think that pen-input interfaces will become very popular in the next few years, even at schools. For students who are novice computer users learning basic mathematics, pen-input with an electronic tablet is a friendlier and easier way of inputting formulas than input with a mouse and keyboard.

For CAI applications in the near future, we designed and developed the Pen-input Mathematical Formula Editor (PenMFE), which is a prototype formula input module with a pen interface.

Generally speaking, it is easy to learn how to use a pen-input interface. As regards formula input, the interface has an additional merit: flexibility of input order. Our prototype module provides users with two important features:

1. Users can input formulas in the same form as when they are writing pen and paper. There are no restrictions on the order in which they write the parts.

2. Users can edit formulas easily and tidy up the layout.

This paper describes the user interface and formula recognition algorithms of the PenMFE.

Overview

The system consists of pen-input hardware and a program.

The hardware consists of an electronic tablet and a stylus (special pen). The tablet is integrated with a liquid crystal display (LCD) screen in such a way that as the stylus slides across the tablet a trace line is displayed simultaneously. The resolution of the LCD screen is 640 by 480 pixels (3 pixels/mm). The scan line density of the tablet is 10 lines/mm and the sampling rate is 67 Hz. The stylus has no cable, so the user is not bothered by it.

The program is the body of the PenMFE. It is called as a subroutine from a CAI application that requires formula input, and is divided into three sub-modules: the Handwriting Recognition (HR) module, the Editing Object Management (EOM) module, and the Formula Recognition (FR) module (see Figure 1). The HR module receives input signals from the tablet and translates them into editing objects (characters and mathematical symbols). The EOM module handles these objects and controls the display on the tablet. After all required objects have been input, the FR module analyzes the structure of a formula and translates it into a character string: a linear form [Nak89]. When the user has completed the input operation, the application that called the PenMFE receives this string and checks it. The linear-form notation represents only the *structure* of a mathematical expression. The PenMFE does not check the mathematical *meaning* of an input formula. Therefore, each application that uses the PenMFE has to have its own parsing routine.

Input of a Formula

Screen Appearance

Figure 2 shows the initial screen of the PenMFE. Most of the screen is occupied by a large blank frame, which is the input area. In this area, a user writes alphanumerics and mathematical symbols with the stylus. He/she can also write some marks and signs, called gestures, which are used in operations for editing formulas.

Below the input area, there are some small frames containing words. These are command buttons. When the inside of a frame is touched with the stylus, the system executes a command.

Free Input Order

In ordinary writing with pen and paper, the order in which mathematical expressions are drawn is not fixed; it depends on personal taste and the user's native language. For example, in the process of writing fractions, some people write a division line first and some write it last. Moreover, most English-speaking people write the numerator before the denominator. On the other hand, most Japanese write the denominator earlier, because it is read first in Japanese. Of course, the notation of fractions is exactly the same in both languages.

The PenMFE also allows a user to input a formula in free order. The user can write characters and mathematical symbols in whatever order he/she likes. The writing position is also free, as long as it is inside the input area. As the user writes objects (characters and symbols), the system recognizes them one by one. Each handwritten object on the tablet is transformed into a letter type or a graphic symbol before the next object is recognized. The user has to write objects at intervals of more than 500 msec, because the system starts the recognition process on receiving a signal indicating that an interval of this length has elapsed since the end of the last writing action.

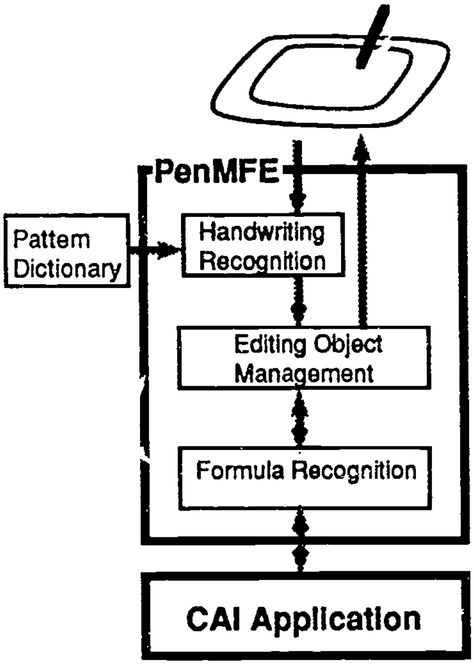


Figure 1. DATA Flow

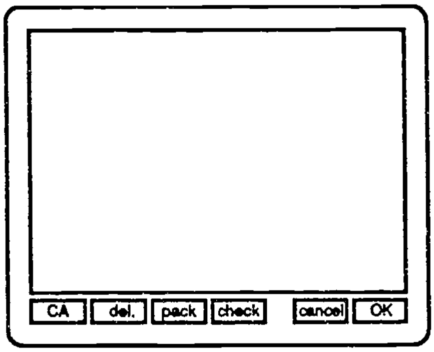
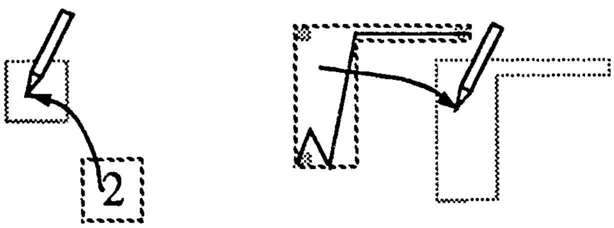


Figure 2. Initial Screen

(a) (b)

Figure 3. Input of Editing Objects



(a) (b)

Figure 4. Movement of Editing Objects

Input of Editing Objects

The transformed characters and symbols are called "editing objects." They are displayed on the tablet and each is enclosed by a frame (see Figure 3(a)). The frame shows the extent of each object. For characters, the size of frames is fixed at 32 by 32 pixels. Even if a user writes a small character, it is transformed into an object of this size. On the other hand, a character should not be written larger than this size, because if it is, the system recognizes it as a mathematical symbol or a gesture. This rule enables the system to distinguish similar objects, such as minus signs and fraction division lines. In the PenMFE (and the former MFE), symbols such as '+','-', and '=' are categorized as characters. Only symbols that have a hierarchical structure are categorized as mathematical symbols. A mathematical symbol always contains at least one character or one mathematical symbol, and therefore no mathematical symbol is smaller than a character.

A handwritten mathematical symbol is transformed into a graphic symbol. The size of this object depends on that of the handwritten symbol. Like a character, a mathematical symbol on the tablet is enclosed by a frame. This frame contains some graphic marks called handles (see Figure 3(b)), which are used for size adjustment.

For instance, if a numerator or a denominator is long, a user can get a long enough fraction division line by drawing a horizontal line of the same length. Even if it is known that the length will be shorter or longer after some other inputs, it is possible to adjust the length by operating the handles.

Editing objects are moveable. If the first stroke of the stylus begins on an editing object (inside a frame), the system recognizes it as a movement operation on the object. The object is moved to the position at which the stroke ends. A user can move editing objects anywhere in the input area by means of this "drag and drop" action (see Figure 4). The size of mathematical symbols can also be adjusted in this manner (see Figure 5).

Adjustment Rules

After all the required editing objects have been input, the user adjusts the position and size of each object, so that the system is able to recognize them correctly as a single formula. There are some rules for the adjustment of editing objects, but no effort is required to learn these rules, because they are almost the same as for mathematical symbols on paper. For example,

Square root: The contents of a square root must be inside the area indicated by the L-shaped frame of the root symbol.

Fraction: The numerator must be above the fraction division line and the denominator must be below the line. Both must be inside borders indicated by the two ends of the division line.

The expression of a superscript (exponent) on the tablet is slightly different from that on paper. It is difficult for a user to write a small character on the tablet, because the resolution of the tablet is rather low in comparison with that of pen and paper. Parallax on the tablet is another cause of this difficulty. For the same reasons, it is also difficult to write characters of different sizes deliberately. Therefore, the size of displayed characters is fixed. A superscript on the tablet is expressed by a gap between the positions at which characters and symbols are located (see Figure 6). If a character (or a symbol) is above the line through

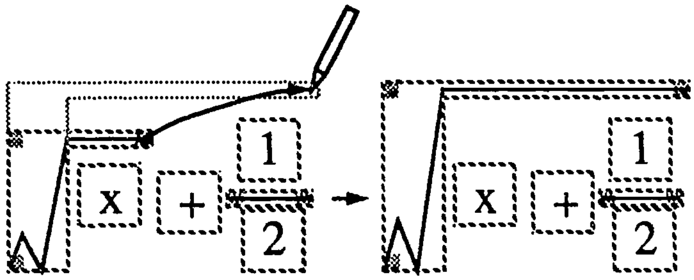


Figure 5. Size Adjustment

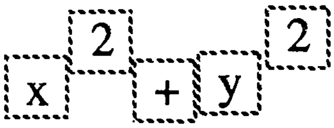
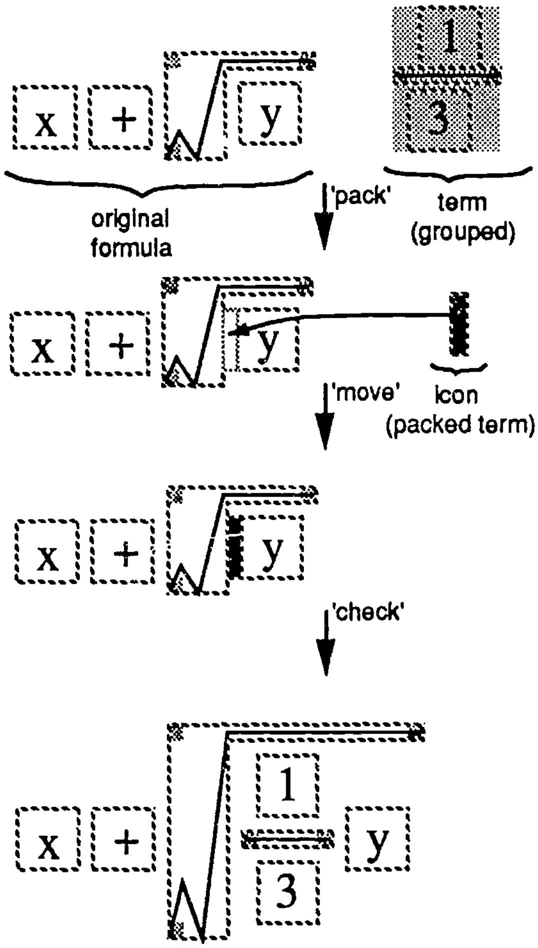


Figure 6. Superscript



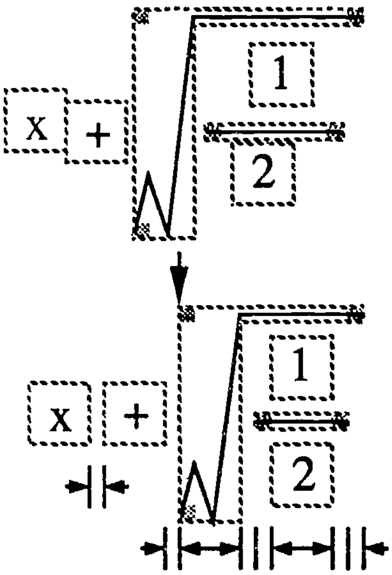


Figure 7. Rearrangement with Spaces

Figure 8. Insertion Operation

the center of a letter to which the character applies, it is recognized as a superscript. When the content of a superscript is not a single character, all of the components must be above the central line. A merit of this method is that multiple superscripts can be used as long as the input area is wide enough.

Editing of a Formula

In the PenMFE system, users can input formulas in the same form as when they are using pen and paper. Furthermore, they can edit formulas. It is easier to modify a formula on the tablet than on paper. Editing objects on the tablet are movable, size-adjustable, and erasable. A roughly formed formula can be rearranged neatly by means of a command. It is also easy to insert a term into the formula.

As mentioned above, editing objects can be moved and resized by means of a "drag and drop" action. Other methods of editing objects include the use of gestures and command buttons.

If a circle is drawn that encloses some objects on the tablet, the color of these objects if reversed, indicating that they are grouped as a single object. These grouped objects can be moved simultaneously.

If an 'X' mark is drawn on an object, the object is deleted from the tablet. By combining these two gestures, "group" and "delete," the user can delete more than one object at a time.

If the 'check' command button is selected, the system checks that the editing objects in the input area form a single formula. If the formula is recognized correctly, it is erased once and displayed again in a neat form. For instance, letters are aligned, symbols are adjusted to the proper size, the numerator and denominator are centered, and so on. In this process, characters and symbols are not placed too close to each other. Proper spaces are placed between them to allow 'insert' operations (see Figure 7).

In ordinary writing, it is troublesome to insert a new term into a formula that is written on paper. On the tablet, however, it can be done easily by combining the 'group,' 'pack,' and 'check' commands. First, a user inputs the new term anywhere in the input area except on the original formula, and uses a 'group' command to group objects that make up the term. When a 'pack' command button is selected, a group is changed into a small icon (see Figure 8). The 'pack' command is available only when the selected objects form a valid term. They have to form a single formula in order to be integrated into the original formula. An icon is a slender symbol that can be moved in the same way as the other objects in the input area. The icon is thinner than the spaces in the rearranged formula, so that it can be inserted anywhere the user wants. The 'check' command is also available for a formula that includes an icon. In this case, the term that is 'packed' in the icon is restored and placed in the original formula.

Algorithm for Formula Recognition

HR module

Stroke data on handwritten characters, mathematical symbols, and gestures are sent from the tablet to the HR module. This module uses pattern matching algorithms for recognition and has a pattern dictionary that covers more than 2000 characters and gestures (most of the characters are Japanese kanji).

However, only alphanumeric and some symbolic data are referred to in the PenMFE system, because the others are not necessary for formula input and editing. As mentioned above, the gesture recognition process starts when intervals of 500 msec occur between writing movements, so characters and symbols must be input one by one.

Formula Recognition Rules

Some methods for formula recognition have been studied [Cha7L, And77]. These involve the the use of a visual language parser that analyzes the mathematical meanings of formulas completely. On the other hand, only structural information on a formula is required in the PenMFE, because a student may input an incorrect or meaningless formula. It is a role of a CAI application to analyze the meaning from the linear form. Consequently, the PenMFE can recognize a formula by using simple algorithms.

The EOM module has a table of editing objects on the tablet, and uses this table to control them. In the table, the following information on each object is recorded: attribute (whether a character or a symbol), type (character code or type of the symbol), position, and size. When a formula recognition process is required in the 'check,' 'pack,' or 'OK' (the end of input) command, the FR module refers to this table and analyzes the structure of a formula. In the first step of the formula recognition process, the left most object is selected. Then, the line through the center of this part is defined as the (typographical) base line of the formula. The center line is a horizontal line that passes through the object's y-axis center. Along this base line, editing objects are checked from left to right and translated into character strings.

When a mathematical symbol appears in the translation process, an area for its content is defined. As mentioned in the previous chapter, the definition depends on the type of the symbol, but the shape of the area is always a rectangle, irrespective of the type. Editing objects in the area are transferred from the original table to a new table. Then another formula recognition process starts with the new table. A user therefore has to arrange these objects so that they are contained completely inside the area.

As a general rule, overlapping of objects on the tablet is not allowed in the formula recognition process, but it is inevitable in the input and editing processes. The results of test use showed that users tend to write characters too close together. This is because the extent of a character spreads when it is transformed from a handwriting into an editing object. For this reason, overlapping among characters is tolerated to a considerable extent in the formula recognition process. Overlapped characters are recognized correctly except when the x positions of the characters are identical.

Discussion

Figure 9 shows samples of mathematical formulas edited by the PenMFE. As these samples demonstrate, the PenMFE can manage most basic mathematical notations that appear in high school mathematics textbooks. The only exception is matrices. The numbers of rows and columns in a matrix are not fixed, so the FR module with its simple algorithms is not able to recognize the structure of a matrix. Another problem is the limited size of the input area, which is caused by the limitation of the tablet's resolution. Because of this, this prototype is not able to handle large formulas such as long equations that consist of many terms and continuously nested fractions. However, this problem will be solved by progress in hardware technology and use of a scrolling feature.

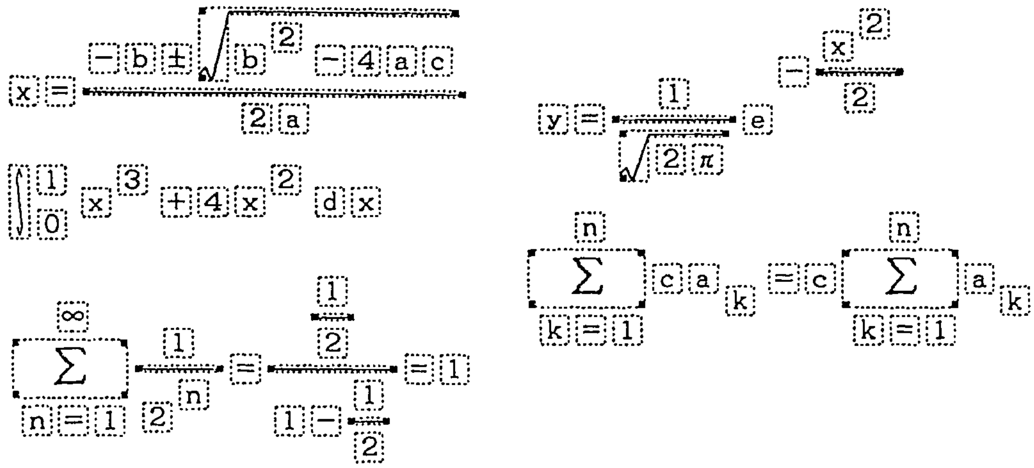


Figure 9. Examples

Conclusion

We have developed a prototype of a mathematical formula input module with a pen interface. The prototype demonstrates the possibility of using a pen interface in CAI applications. We are planning some experiments to measure the usability of the PenMFE. Improvement of some features and integration of the PenMFE into our CAI application are also planned.

References

[Aka92] M. Akaishi, Y. Nakayama, et al., *Mathematical Software as an Exploration Tool*, Proceedings of NECC '92, pp. 145-151, National Educational Computing Conference, 1992.

[And77] R. Anderson *Two-Dimensional Mathematical Notation*. In K. Fu, editor, Syntactic Pattern Recognition Applications. Springer-Verlag, 1977.

[Cha70] S. K. Chang, *A Method for the Structural Analysis of Two-Dimensional Mathematical Expressions*, Inf. Sci. 2, pp. 253-272, 1970

[Hid92] K. Hidaka, *Development of GeoBlock: a Micro-World for Learning and Teaching Geometry*, Lecture Notes in Computer Science, No. 602, pp. 294-306, Springer-Verlag, 1992.

[Nak89] Y. Nakayama *Mathematical Formula Editor for CAI*, Proceedings of ACM CHI '89, pp. 387-392, 1989.

Teaching Science Laboratory Classes with Multimedia: An Australian Experience.

MICHAEL W. NOTT, MATTHEW D. RIDDLE & RICCI E. SWART*
*Faculty of Science Multimedia Teaching Unit and *Television and Optical Disc Development Unit,*
The University of Melbourne, Parkville,
Victoria, Australia 3052

Interactive computer simulations of science experiments are increasingly common. Such programs are potentially popular with students and staff as substitutes for laboratory exercises. But there are two major problems: firstly, computer simulations usually do not show details of the experimental set-up. Such details are important if the program is to be maximally useful for teaching because skills acquisition is an important aspect of practical laboratory classes. Secondly, such simulations generally rely on mathematical models of actual experiments and, as such, they deprive students of the opportunity of working with real experimental data, which is a critical factor in science education.

The general aim of the project was to place a previously text-based course in laboratory pharmacology within a Multimedia environment (drawing on video, sound, simulations, animations, illustrations, still photos, chart and statistics packages all on the one digital display screen). Firstly this involved producing QuickTime™ videos which would better show how to set up experiments than did the original protocols. Secondly, the same videos were also to be used to introduce existing computer-based simulations so that the simulations became more realistic and effective in teaching. Thirdly, in order to allow th- students to work with actual data, without setting up the experiments, a series of analogue to digital records of actual experiments were generated using MacLab™ (ADInstruments, Australia) hardware and software. HyperCard™ is used as the front end for the overall student-computer interaction. At all stages of the project close consultation with research groups in the Pharmacology Department was encouraged, so that much of the actual experimental data is to come from the research laboratories.

Ancillary objectives of the project were to: integrate our current and developing Multimedia resources into a single teaching package, design, and evaluate the package with two key objectives: enhanced learning and more efficient teaching, bring the program to the point where it may be published as a digital compact disc for use in Dep... f Pharmacology throughout Australia and internationally, and to provide a model ... ciation in the development and use of Multimedia for teaching in other University dis... The Australasian Society for Clinical and Experimental Pharmacology and Toxicology offe... co-ordinate thses activities.

The current report is based on student laboratory exercises in pharmacology (Nott, Thaina & Riddle, 1989a, 1989b; Nott, Thaina, Riddle & Swart, 1992a, 1992b, 1992c) but the system is a model for other science disciplines. Videotaping, editing and audio mixing were carried out by the University of Melbourne Television and Optical Disc Development Unit. Seven 3 to 5 min video sequences were compiled to make them appropriate 'triggers' for the computer simulations. Each trigger was further edited down to a 30 sec to 1 min duration to minimize file size requirements for QuickTime™ sequences.

Computers in student practical laboratories

Many of the experiments conducted in Pharmacology and Toxicology (and in other biosciences) involve the measurement of muscle contraction, respiratory function, vascular activity or heart, brain or nerve depolarization. Typically, change of muscle length, development of force, changes in blood pressure, change in resistance to air flow or changes in electrical activity are measured using appropriate transducers and amplifiers. In the past, the output of these transducers and amplifiers would be recorded on oscillograph analog pen recorders. But recently computers have been introduced for this task. The system presently being used consists of Macintosh™ computers receiving signals from MacLab™ analog to digital converters. The MacLab™ can convert up to 8 analog signals simultaneously and MacLab™ software has been written to easily handle the Macintosh™ side of the workstation. The system is user friendly and requires only an hour or so of tutoring at the beginning of the Semester before students are able to start using it.

In Physiology and Pharmacology at The University of Melbourne there are 64 MacLab™/Macintosh™ IIci workstations (Pharmacology has 16 of them; Fig 1, left). They are networked into 4 systems, each served through Ethernet by a Macintosh™IIfx and laser printers. The Macintosh™IIfx is connected to remote University systems by optical fibre. The display units at each workstation also support video, and the laboratories are wired for video transmission (Fig 1, right). Thus, not only do the systems function as recorders but are also well suited as Multimedia tutorial devices. So the students, while engaged in a practical class, also work through associated materials so that the practical class becomes a rich learning environment incorporating videos, sounds, simulations, graphics, still photos and text, graphical and statistical recording packages.

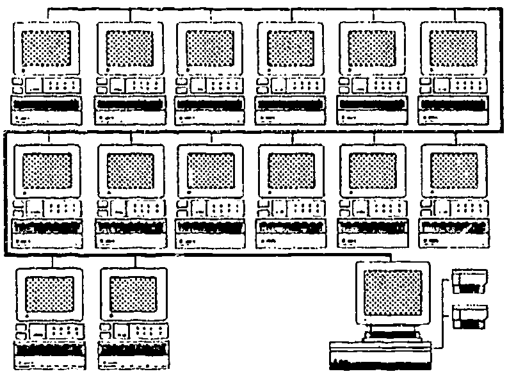



Figure 1 The Department of Pharmacology multimedia-capable laboratory
Left: Multimedia systems in the Department of Pharmacology, The University of Melbourne. The workstations function as data acquisition devices through the MacLab™ analog-digital converters, and as Multimedia systems served by videotape videodisc and CD-ROM. Student reports are transferred to the Macintosh™IIfx server for assessment and feedback by staff.
Right: Students using the MacLab™/Macintosh™ Multimedia workstations in the Department of Pharmacology, The University of Melbourne.

Integration into existing courses

The Program is a resource which can be incorporated easily into existing courses. The Pharmacology Program will replace and enhance existing laboratory protocols. Our surveys indicate that, as in our own department, many other departments will use the Pharmacology Multimedia Practical Program 'as is' — simply running their practical laboratory classes according to protocols and resource materials as they appear in the Pharmacology Program. Alternatively, the Pharmacology Multimedia Practical Program can be used as an adjunct to established course material. In this respect the Program can be valuable to departments wanting to run it alongside their existing practical curriculum as a computer-based tutorial system. It may also be used to supplement teaching in lectures, where examples and data can be called up for screen projection. For maximum flexibility, simple generic software tools are used in order to allow the updating or modification (according to the needs of a particular department) of textual material into practical protocols and tutorials.

A typical experiment in Pharmacology is one which examines the effects of a drug, acetylcholine, on the smooth muscle of the intestine. The experiment illustrates some of the important aspects of pharmacology laboratory work: the skill of setting up a biological preparation, the recording of responses (in this case contractions of the smooth muscle), the measurement of responses, the graphing of intensity of response against concentration of a drug, the nature of variation between responses, the application of statistics to the results, the ability to write reports and draw appropriate conclusions.

Text (on screen) retained as basis of Multimedia protocols

At present the older style written practical laboratory manual is used as the basis for the course. But the protocols have been precised for screen display. Nevertheless, on-screen text remains the primary instructional medium. Each experimental protocol appears with the typical sections — Introduction, Methods, Results and Discussion — which together constitute the well established structure of scientific journal papers to which students are exposed early on their course work. It is envisaged that there will later be electronic journals for Pharmacology (and other sciences), to which our students will eventually contribute, and that these journals may have their own editorial style. But we predict that the basic structure of the scientific paper (noted above) will be retained, and that on-screen text (albeit liberally sprinkled with QuickTime™ video and audio) will be the key instructional medium. But the appropriateness of on-screen text needs to be determined. It is therefore planned that for at least one of the practical protocols a comparison will be made between three different modes of screen presentation, in which the following modes of instruction carry the protocol content: text based instruction, coupled with QuickTime™ video sequences (called up on demand by the student); point by point instruction; coupled with QuickTime™ video sequences (called up on demand by the student); point by point instruction (as above) with voice over, coupled with QuickTime™ video sequences (called up on demand by the student).

In the *Introduction* to each practical protocol, the student reads the background and the reason for doing the experiment. But students are expected to know the background to the experiment and the reasons for using the particular experimental protocol. A novelty introduced here is an electronic 'Demonstrator'— an icon of a face which appears on the screen periodically, throughout the whole program, to interrogate the student, using voice or text, about the work they are immediately engaged in. The 'Demonstrator' may be easy (friendly 'Tutor' icon) or tough ('Professor' icon). 'Demonstrator' questions come from a HyperCard™ stack.

With the student having a firm knowledge of the *Introduction* to the experiment, he or she moves on to the *Methods* section in which is concentrated the Multimedia sequences, particularly QuickTime™ videos. If, for example, the student needs to see or hear how biological tissue is set up for recording they click the mouse on the appropriate hot-spot in the text or refer to a pull down-menu. Similarly they call up in QuickTime™ other key parts of the Methods (Figure 2).

At any time the student can refresh their memory about a technique by calling up the appropriate QuickTime™ video from a menu comprising titles of all the QuickTime™ sequences in the protocol. Then, when the student has set the preparation up (or, in the case where the student is working with prerecorded data or a simulation, has been appropriately orientated with the procedure) the *Results* section begins with the application of drugs and the recording of responses. In the practical course program the students spend one laboratory session using pre-recorded data or a simulation and then, in the next session, to do the actual experiment.

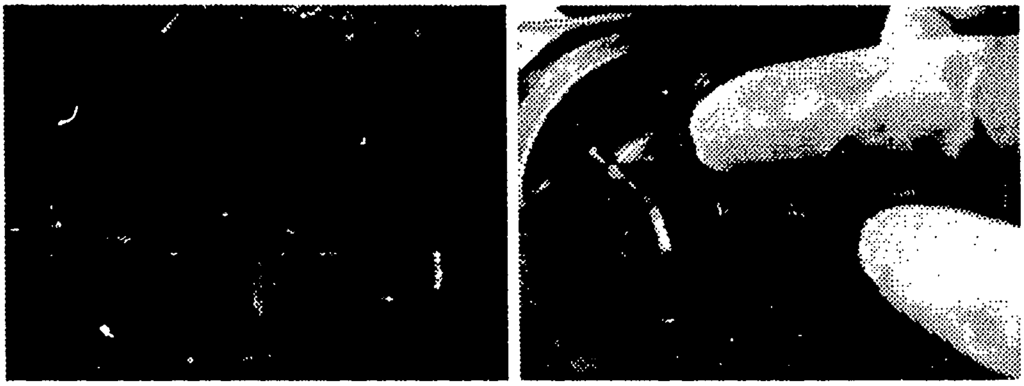


Figure 2 Two QuickTime™ sequences (actual size)
Left: the correct set up for an experiment using an organ bath preparation for measuring contractions in a length of intestine from the guinea pig, and right: a demonstration of how to adjust the tension on the tissue.

Finally the student is ready to write their report as though it were a *Discussion* in a scientific paper. They construct the report on the computer using text, graphs and other illustrations from their experiment. In the *Discussion* section the student may be asked (by the icon 'Demonstrator') a number of key questions, and thus be urged to draw conclusions relating to the experiment. The student's report is sent back to the server Macintosh™IIfx for subsequent marking by the teachers. The Macintosh™IIfx is linked to laser printer so that the student may get a hard copy of their results if they so wish.

Increasing the effectiveness and efficiency of laboratory teaching

Teaching science through laboratory work is essential but is expensive in terms of staff and materials. With increasing student numbers, many students who wish to take Science Practical Classes are not able to be accepted. For these reasons the Department of Pharmacology, like other science departments, is constantly seeking to make its laboratory based teaching more efficient. In 1991, when we upgraded our practical class laboratories with computer-based data acquisition systems, we took the opportunity to make a major impact on our effectiveness and efficiently by ensuring that the systems were optimal for delivery of interactive Multimedia. Features described in Table 1 suggest that the program should enhance learning, enhance skills acquisition and increase the efficiency of teaching (i.e., less time required by students and staff to accomplish particular exercises).

Table 1

Features of the Multimedia Program and intended outcomes

Feature	Intended outcome	
Replace traditional print practical protocols with flexible and more engaging Multimedia 'on screen' schedules.	enhanced learning	
Systematize information delivery using Multimedia and therefore minimize variation between tutors.		increased efficiency
Reduce the time taken by tutors to deliver trivial information and free up time to engage in deeper learning.	enhanced learning	increased efficiency
Allow the option of running practicals as simulations, as recall of prerecorded experiments, or as actual experiments.	enhanced learning	increased efficiency
Incorporate computer-based spreadsheet and graphing exercises into practical schedules.	enhanced skills	
Introduce, as simulations, new practical exercises that are otherwise too complex or ethically undesirable to conduct.	enhanced learning	
Save on the use of some animal preparations, again by running simulations — ethically desirable.		increased efficiency
Introduce students to exemplary post-graduate scientists through biodata and key experimental findings.	enhanced learning	
Encourage the writing of reports using Multimedia, thereby introducing students to the concept of electronic publishing and making 'on-screen' assessment of reports possible.	enhanced skills	increased efficiency

Evaluation

For each feature of the Pharmacology Multimedia Practical Program shown in Table 1, intended outcomes will be monitored and evaluated, appropriately, in terms of: patterns of use according to previously installed computer-based log; responses in student interviews; questionnaires of staff and students; achievement data.

In the development and testing period of the project most emphasis has been placed on a course in Pharmacology to Second Year Science students: 28 students spend 3 hours per week for 13 weeks in the practical laboratory. Comparisons will be made between report marks and final exam results of a 1993 cohort and those of students taking the same course over the preceding 3 years. Comparisons will be made between performances of the cohort in 1993 classes which are run on Multimedia and in control classes which are run using traditional print protocols with no Multimedia.

Development activities

In sequencing the various activities for the development of the program it was necessary to order activities as follows: brief Designer/Programmer; transfer and adapt practical schedules to computer; customize Microsoft™ Excel™ spreadsheets with statistical macros & graphing (Excel™ adopted because it is the most common spreadsheet package); assemble 100 video, sound and animation sequences in QuickTime™ format (QuickTime™ is free with all colour-capable Macintosh™ computers); scan and animate 34 simple line drawings to show the key point of each practical; design HyperCard™™ for management & interactive software system (Comes free with all Macintosh™ computers and is extremely versatile Multimedia management package); develop 'Demonstrator' — an icon of a face which appears on the screen periodically to interrogate the student, using voice or text, about the work they are engaged in. The 'Demonstrator' may be easy

ERIC Full Text Provided by ERIC

— friendly Tutor icon; or tough — Professor icon. 'Demonstrator' draws on the HyperCard™ stacks such as profiles of drugs used in pharmacology and toxicology practical laboratories, formulae of physiological saline solutions, references, pharmacological anecdotes including profiles of recent graduates, multiple choice questions (MCQs; courtesy University of Sydney, Department of Pharmacology). Further activities were to develop MCQ scoring routines; trial and tune the program in Science practical classes, demonstrate/discuss the program at national and international learned society meetings and prepare reports for publication. Final activities are to advertise and publish the program as CD-ROM.

Development software

The central development software chosen for this project is HyperCard™, which has become a widely used environment for implementation of educational software on the Macintosh. HyperCard™ allows the combination of textual material with video, sound, animation and graphics in a highly interactive way, and provides the necessary tools for the design of interfaces which are clear and easy to use. Apple Computer's recently-released QuickTime™ format for time-based data such as video made practical the integration of short video sequences. Control of QuickTime™ sequences using HyperCard™ can be achieved using external commands (XCMDs) provided by Apple™ for this purpose. These XCMDs allow video sequences to be initiated by HyperCard™ scripts, and any on-screen action within HyperCard™, such as a mouse-click on a button or a menu selection.

An engine for recall and playback of digitized data

A software engine for the recall and playback in real time of data recorded using the MacLab™ analogue-digital recording system has been developed. This MacLab™ Engine allows students to call up actual experimental data, in graphical form, from within HyperCard™, as though they were conducting the experiments themselves. A response such as, for example, an increase in blood pressure is represented as a developing trace scrolling in real time across the screen. The student user has the ability to navigate through the experiment, analysing results as he or she does so. This means they have access to a large and expanding bank of experimental data built up by researchers at the Department of Pharmacology, as well as to data they collect from their own experiments.

Video format

Video sequences were recorded and edited using BVU-SP broadcast quality video tape. The sequences were shot with a light blue background and high contrast lighting to improve clarity and luminescence, which were seen as particularly important because of the small size of the final QuickTime™ window. This factor combined with the low frame rates in comparison to traditional PAL video display (25 frames per second; fps) imposed certain restrictions on video content, making the inclusion of extreme wide shots and fast pans and zooms undesirable. Dissolve transitions were preferred to direct cuts. Talking heads were avoided because of difficulties with accurate lip-synchronization at low frame rates.

The sequences were later digitised using a VideoSpigot™ NuBUS video capture board made by SuperMac on a Macintosh™ IIci computer. It was possible to capture smooth PAL video sequences at 10 fps at the window size of 192 x 144 pixels (see Figure 2). All sequences were then compressed using the QuickTime™ video compressor.

ERIC

Currently we are able to play back the QuickTime™ video sequences at the rate and frame size at which they were captured, in 16 bit (near photographic quality) colour. Limitations imposed by hardware on display of video material using QuickTime™ involve, among other factors, data transfer rates of storage devices and the speed of the Central Processing Unit of the computer. The three major software factors involved in digital video display are image quality, frame size, and frame rate of the video sequence. As a general rule, varying one factor will have an inverse effect on at least one of the other two, so that, for example, increasing frame size will reduce playback frame rates and necessitate a reduction in quality of the image. Because of the large file sizes generated using digital video, sequence lengths were kept to a minimum. Despite the current limitations imposed by digital video, it was possible to include high quality digital video in an interactive way using HyperCard™.

Conclusions

The authors have vetted the needs of pharmacologists and toxicologists for this type of interactive Multimedia program to teach student laboratory classes and then insisted on professional production standards in terms of content, video, sound, graphics and interface design. In so doing it is hoped to produce programs that result in more interesting and engaging pharmacology and toxicology laboratory sessions. The work is relevant to other disciplines introducing simulations and Multimedia to their students. It could encourage more emphasis on the laboratory environment as the most appropriate for learning science. It could introduce students to electronic publishing. Issues that are of concern, and which will be investigated, are whether the traditional research paper format should be the protocol metaphor and whether text (on screen) should remain the basic structure of the practical class protocol.

Acknowledgments

The authors acknowledge grants from New Educational Aids in Medicine and Science (Australia) and The University of Melbourne's Department of Pharmacology and The Centre for the Study of Higher Education. We are grateful for continuing assistance and advice from the Interactive Multimedia Learning Unit, The University of Melbourne.

References

Nott, M.W. & Riddle, M.D. (1989a) Teaching drug screening procedures using and interactive videodisc/Macintosh™ system. Proceedings of the Australian Physiological and Pharmacological Society, 20, (1), 94P.

Nott, M.W., Thaina, P. & Riddle, M.D. (1989b). Teaching practical pharmacology using the MacLab™/Macintosh™ computer system. Proceedings of the Australian Physiological and Pharmacological Society, 20, (1), 95P.

Nott, M.W., Thaina, P., Riddle, M.D. & Swart, R. (1992a). A videodisc and interactive computing system for teaching practical pharmacology. Clinical and Experimental Pharmacology and Physiology, Supplement 18, 45.

Nott, M.W., Thaina, P., Riddle, M.D. & Swart, R. (1992b) Seven videotapes, a videodisc and an interactive computing system for teaching practical pharmacology. Proceedings of the Australian Physiological and Pharmacological Society, Abstract in Press.

Nott, M.W., Thaina, P., Riddle, M.D. & *Swart, R. E. (1992c). Interactive teaching of science practical work using recall of digitally recorded data and QuickTime™ video. Proceedings of the Australian Society for Computers in Learning in Tertiary Education, Sydney, Australia, 168 - 172.

ERIC

Hypermedia-based Interactive Student-Assessment System (HISAS): Concept and Architecture

JODY PAUL
Department of Computer Science and Engineering
University of Colorado at Denver, Denver CO 80217-3364
Internet: jody@cudnvr.denver.colorado.edu

The Hypermedia-based Interactive Student-Assessment System (HISAS) is a new concept that involves the integration of hypermedia with key requirements for advancing students' education. It includes creation of an interactive system for assessing student competence that can foster individualized progress through a given curriculum and provide instructors and administrators with quantitative, standardized assessments. In this paper, we describe the rationale, concept and architecture of a HISAS. We then present preliminary results of our prototype development effort.

Assessment: Value and Difficulties

Assessment serves a variety of purposes, central among which is to support decisionmaking that concerns students' scholastic, personal and social characteristics, instructional progress, and institutional adjustments. These decisions typically involve problem diagnosis, planning and conducting instruction, and official assessments such as grading. Assessment methods can enhance the educational experience when used as training devices, as techniques for evaluation, and as guides for instructors and administrators in curriculum design and teaching methods.[Airasian 1991]

There is increasing governmental and administrative interest in the potential contribution of comparisons of pre- and post-experience evaluations ("value-added" assessments of the educational process) and significant concern over how assessment can be used to "foster intellectual development and educational success."[O'Connor 1992] Resnick and Resnick report that the nature of standardized tests now widely used for accountability are "fundamentally incompatible" with the goal of improving students' higher-order abilities (i.e., problem solving and thinking) and that alternative assessment methods are needed.[Resnick and Resnick 1992] There is a widely recognized need for a "reformulation of assessment to help not hinder the effort to teach thinking."[Shepard 1992]

Direct contributions to the educational experience of individual students come from both the assessment process itself and the resulting feedback. The more involving the assessment process and appropriate the response latency, the more effective is the exercise.[Dworkin and Dworkin 1988] A review that includes explanations of *why* answers were incorrect and likely *sources* of incorrect reasoning is more useful than simply "correct/incorrect" responses.[Kellaghan, Madaus et al. 1982] In addition, a much-valued method in the arts, sciences and trades is *performance testing* which evaluates a student's ability to solve problems that require the performance of particular tasks or procedures.[Sweezey 1981; Priestley 1982; Airasian 1991]

The primary difficulties with achieving these *involvement*, *feedback* and *performance* goals are the prohibitive resource requirements necessary to accomplish them using standard assessment tools and environments. Achieving immediate, customized feedback requires an impractical student-examiner ratio. Providing all students with complete, controlled task environments is often costly (e.g., expensive instruments or materials), sometimes dangerous (e.g., hazardous materials or dangerous procedures), or otherwise infeasible (e.g., hypothetical or extra-terrestrial environments).

The availability of hypermedia-capable interactive computer systems, however, suggests alternative means to achieve these goals. Such systems can present the problems (simulating a task-environment if necessary), accept student responses, evaluate those responses, and present customized feedback in real time.

The HISAS Concept and Architecture

The HISAS concept blends hypermedia with techniques from education, artificial intelligence and human-computer interaction to provide students with an automated assessment tool that responds dynamically to their particular needs and situation.

Hypermedia, the organization of a multiply-connected set of text, sound, graphics and animation (dynamically-evolving images), offers unique advantages for advancing students' education. The subject-specific collection of multimedia elements permits nonsequential processing of knowledge by specifying connections between the component parts of the medium.[Jensen and Nissen 1988] This aspect extends the involvement of the student (e.g., by bringing additional senses into play and employing both "left-" and "right-brain" faculties), expands the set of applicable domains (e.g., including both chemistry and music appreciation), and provides a unique opportunity for practicable performance-testing.

The interactive nature of a HISAS makes it a participatory, rather than static, assessment tool. A HISAS can provide timely feedback, unlike conventional examinations where the student reads questions and writes answers which are later evaluated and, even later, the results presented to the student. (We note that providing immediate feedback is not always the best pedagogical choice. The HISAS architecture allows complete flexibility with respect to the immediacy or latency of feedback.) This feedback includes explanations specifically tailored to the performance of the student. In this regard, we have extended work in multipolar-testing and confidence-measuring assessment to exploit the discovery of students' states of information (informed, partially informed, misinformed, uninformed) and enable the system to provide corrective and directive advice that addresses typical misunderstandings associated with a student's particular pattern of responses. For example, if a student makes a "very wrong" response, remedial support may be indicated unless a common misconception is identified from the pattern of other responses. A student who picks midway between the correct answer and another plausible answer may simply require some conceptual fine-tuning. As the HISAS concept evolves, this relatively simple yet powerful strategy will be extended, enabling the system to adapt once it has learned more about the student through direct interaction or through intercession by a human agent (e.g., a mentor).

Along with the use of the computer to provide timely, customized feedback to the student, a HISAS increases student involvement by using multiple media, including tactile (e.g., keyboard and mouse), aural (e.g., voice and non-voice sounds), and visual (e.g., texts, diagrams, and animations). Involvement is further enhanced by discourse knowledge that enables a HISAS to respond to the particular student in the active context.

A HISAS also provides the ability to do performance assessment, in which the student is given a task to perform in a "virtual world" provided by the HISAS. For example, where physical observables must be simulated (e.g., for a chemistry procedure in which the student has just virtually combined solutions A and B) an animated sequence may be displayed to show the results (e.g., showing the progress of the interaction of the solutions as it would occur in real time). The use of such simulation is indicated when actual performance tests are impractical, due to cost, danger, the serious consequences of mistakes, or the impossibility of arranging actual performance situations. By simulating performance conditions, the HISAS controls most of the variables in the testing situation and can standardize the assessment across candidates and administrations.[Priestley 1982]

Foundations

The development of a HISAS depends on diverse theoretical and technological foundations. A well-integrated hypermedia support-base provides the desired multimedia capability coupled with the ability to link and navigate through domain and pedagogical information. Hypermedia techniques support initial training in the use of the system, facilitate students' use of the system, enhance the presentation of a problem's context, and provide the human-interface to the simulated environment used for performance assessment.

A solid understanding of the pedagogical process regarding the field of application is essential, including models for effective interaction and learning. The predominant contributions from the field of education are the essentials of assessment, principles leading to the design of beneficial feedback, and the basic notions concerning performance testing.[Sweezey 1981; Kellaghan, Madaus et al. 1982; Priestley 1982; Dworkin and Dworkin 1988; Freedle 1990; Airasian 1991; Gifford and O'Connor 1992; Sternberg 1992]

Notions of discourse and discourse-modeling from natural-language processing are incorporated to enhance student–HISAS interaction.[Grosz 1978; McKeown 1983; McKeown, Wish et al. 1985] At present, we use a simple model of discourse, including the ability to maintain dialog context. That is, student actions are interpreted in the active context, so identical actions may have different interpretations as appropriate to the current state of the interaction.

Our design is influenced strongly by the student-modeling work done in the field of intelligent tutoring systems (ITS).[Nwana 1990; Wood and Holt 1990; Murray and Woolf 1991] A significant, and fundamental, distinction between HISAS and general work in ITS is the objective of student assessment, not of teaching a curriculum. A HISAS may prove to be a useful component of an ITS or a tool for ITS researchers to evaluate the effectiveness of their systems.

A HISAS needs to modify models of individual students dynamically and in real time. In our initial design, we incorporate skeletal, generic student-models.[Freedle 1990; Murray and Woolf 1991] As we assimilate the results of analyzing field trials, our models will be extended to include the identification and tracking of students' styles, goals, and plans.

ERIC

The HISAS Feasibility Prototype

The prototype HISAS uses multimedia presentations and provides feedback to the student that includes corrective and directive suggestions. It uses a general model for assessment based on confidence-measuring procedures and multipolar testing. The student model is generic and the discourse model is based on sketchy scripts.

A hypermedia-based training phase is provided to prepare students for the actual assessment process. During this phase, students are tutored in the use of the HISAS and become acquainted with the multipolar confidence-measuring method. They practice using the system directly and gain understanding of the implications of their responses. During actual testing, the HISAS uses integrated multimedia to present the problem context and feedback.

The basic assessment model centers on the use of 3-way multiple choice questions and a 16-region response template. Figure 1 shows a simplified layout of a sample question-response presentation (actual presentations use color, shading, sound and action sequences as appropriate). The values awarded for each selectable region in the case where the single correct answer is "A" are shown in Figure 2. (Scoring is symmetric with respect to each vertex for correct answers of "B" or "C".)

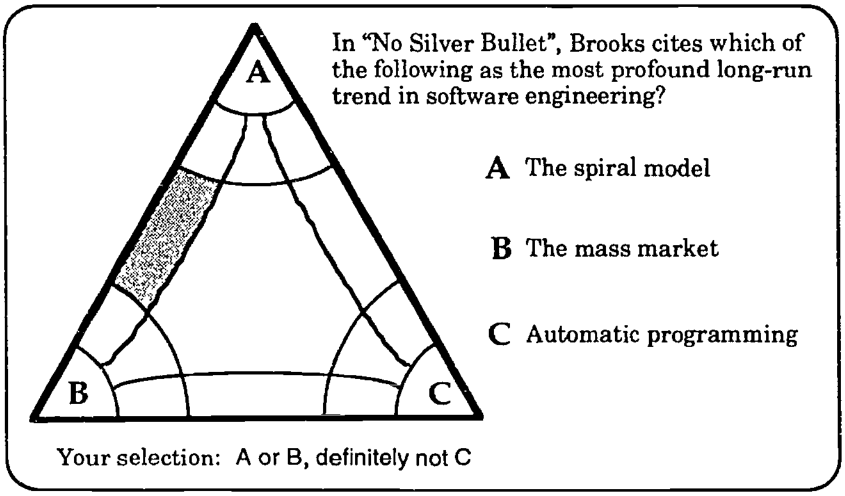


Figure 1. Sample question-response presentation.

The HISAS continually provides visual and textual feedback to the student. Color, shading and animation are used to facilitate students' understanding of the selection template. Textual confirmation is given as to the interpretation of the selected region. Flexibility in providing scoring information allows the examiner to decide *a priori* when and how that information should be presented to the student (e.g., only at the student's request, immediately upon committing to an answer, immediately upon completing the entire exercise). Hypermedia excursions are provided to allow students to explore the implications of their selected responses, including suggestions for improving understanding and future performance.

The prototype also includes a relatively primitive task-performance assessment example. Figure 3 shows the sample task-performance display and interaction. Here, the

ERIC

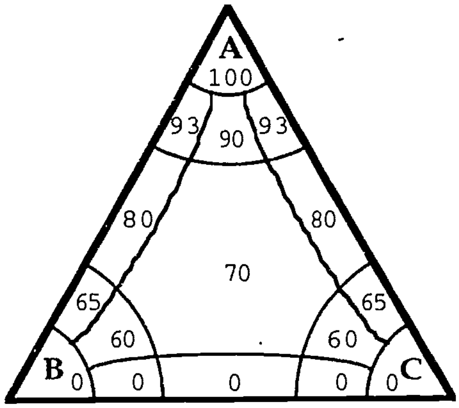


Figure 2. Confidence measuring response template, showing values awarded for each selectable region in the case where the single correct answer is "A".

student has observed a customer explaining his desires, and is in the process of developing a top-level data-flow-diagram (the context diagram) using the palette of available objects.

Examiners may engage in an interactive dialog with the HISAS that facilitates entry of new problems. A full spectrum of levels of initiative-sharing between the HISAS and examiner is available. This supports examiners who desire complete flexibility in problem design, presentation, scoring and feedback, while also supporting those examiners who prefer to be prompted for the necessary information and to accept standardized templates.

Our initial prototype addresses university-level software-engineering (SWE) education, focused on life-cycle models and requirements engineering. The methods and techniques developed for and employed in the prototype are in no way confined to the SWE domain of application. The SWE test-bed was chosen because it allows us to rapidly conduct experiments with minimal effects related to subjects' familiarity with the

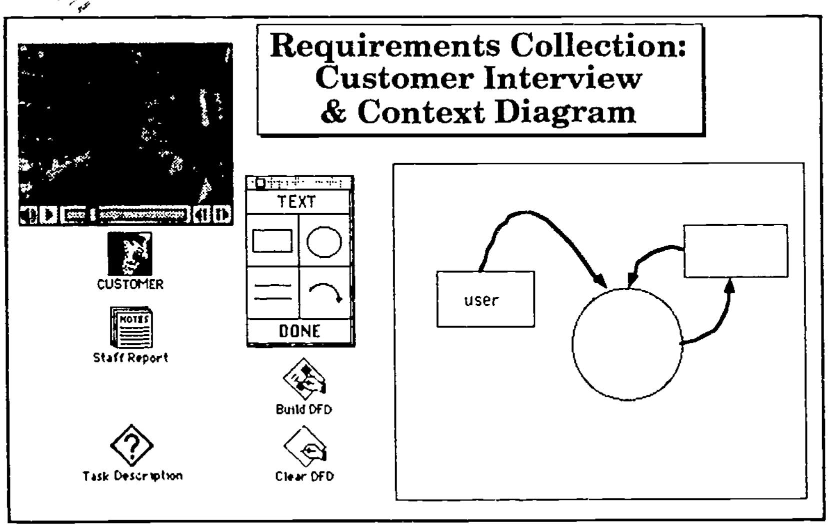


Figure 3. Sample task-performance environment.

implementation technology, such as lack of experience using a keyboard or mouse or inherent fears of using a computer.

Next Steps

We are currently engaged in field tests of the initial prototype. In addition, we are developing a more substantial system targeting high-school-level chemistry curriculum, extending the performance-testing aspect, and improving the student and discourse models.

A key issue that arises when attempting to assess the task-performance of a student concerns *what to measure.* We could choose to look only at the result achieved, but that information may not be sufficient. (For example, it may be possible to achieve the same result by inappropriate means, such as using quicksort when binary sort is requested.) Alternatively, we could look at the specific command sequence issued by the student, but there may be an infinite variety of functionally-equivalent sequences. Our approach, instead, is to look at intermediate-goal satisfaction, which presumes a sequence of primitives that achieve necessary sub-goals of the major goal.

References

Airasian, P. W. (1991). *Classroom Assessment.* New York: McGraw-Hill.

Dworkin, A. & Dworkin, N. (1988). *Problem Solving Assessment.* Novato: Academic Therapy Publications.

Freedle, R., Ed. (1990). *Artificial Intelligence and the Future of Testing.* Hillsdale: Lawrence Erlbaum Associates.

Gifford, B. R. & O'Connor, M. C., Eds. (1992). *Changing Assessments: Alternative Views of Aptitude, Achievement and Instruction.* Boston: Kluwer Academic Publishers.

Grosz, B. J. (1978). Discourse Analysis, in *Understanding Spoken Language*, D. E. Walker, Ed. Elsevier North-Holland. 235-268.

Jensen, J. A. & Nissen, T. (1988). Paedagogik, Psykologi, Hypermedier: Hypermediers paedagogiske potentiale i erkendelsesteoretisk og kognitionsteoretisk perspektiv. *Psyke and Logos* , 9(2), 419-442.

Kellaghan, T., Madaus, G. F. & Airasian, P. W. (1982). *The Effects of Standardized Testing.* Boston: Kluwer-Nijhoff Publishing.

McKeown, K. (1983) *User-Oriented Explanation for Expert Systems.* Department of Computer Science, Columbia University. CUCS-85-83.

McKeown, K. R., Wish, M. & Mathews, K. (1985) *Tailoring Explanations for the User.* Department of Computer Science, Columbia University. COCS-172-85.

Murray, T. & Woolf, B. P. (1991). A Knowledge Acquisition Tool for Intelligent Computer Tutors. *SIGART Bulletin* , 2(2), 9-21.

Nwana, H. S. (1990). Intelligent Tutoring Systems: An Overview. *Artificial Intelligence Review* , 4(4), 251-277.

O'Connor, M. C. (1992). Rethinking Aptitude, Achievement and Instruction: Cognitive Science Research and the Framing of Assessment Policy, in *Changing Assessments: Alternative Views of Aptitude, Achievement and Instruction*, B. R. Gifford and M. C. O'Connor, Ed. Boston: Kluwer Academic Publishers. 9-35.

Priestley, M. (1982). *Performance Assessment in Education and Training: Alternative Techniques*. Englewood Cliffs: Educational Technology Publications.

Resnick, L. B. & Resnick, D. P. (1992). Assessing the Thinking Curriculum: New Tools for Educational Reform, in *Changing Assessments: Alternative Views of Aptitude, Achievement and Instruction*, B. R. Gifford and M. C. O'Connor, Ed. Boston: Kluwer Academic Publishers. 37-75.

Shepard, L. A. (1992). What Policy Makers Who Mandate Tests Should Know About the New Psychology of Intellectual Ability and Learning, in *Changing Assessments: Alternative Views of Aptitude, Achievement and Instruction,* B. R. Gifford and M. C. O'Connor, Ed. Boston: Kluwer Academic Publishers. 301-328.

Sternberg, R. J. (1992). CAT: A Program of Comprehensive Abilities Testing, in *Changing Assessments: Alternative Views of Aptitude, Achievement and Instruction*, B. R. Gifford and M. C. O'Connor, Ed. Boston: Kluwer Academic Publishers. 213-274.

Sweezey, R. W. (1981). *Individual Performance Assessment: An Approach to Criterion-Referenced Test Development*. Reston: Reston Publishing Company, Inc.

Wood, P. H. & Holt, P. D. (1990). Intelligent Tutoring Systems: An Annotated Bibliography. *SIGART Bulletin* , **1**(1), 21-42.

Acknowledgments

Special thanks to Allen Klinger (U.C.L.A.) for his contributions to the work on confidence-measuring and the evolution of the HISAS concept.

HELENA Hypermedia Project: from Knowledge Representation to Knowledge Construction

VALERY A. PETRUSHIN
Intelligent Tutoring Systems Lab.
V.Glushkov Institute of Cybernetics Ukrainian Academy of Sciences
40, Prospect Akademika Glushkova
252207 Kiev, The Ukraine
E-mail (Internet): petr%itslab.kiev.ua@relay.ussr.eu.net
or petr%itslab.kiev.ua@fuug.fi

INTRODUCTION

Learning environments are used to supply a learner some computer resources for supporting learning process. In contrast to tutoring systems learning environments as a rule have no information about instructional objectives and problems being solved by the learner. A learning environment supplies a learner with at least one of the following resources: 1) information about the structure and contents of subject being learned and 2) tools for constructing and exploring computer models such as drawings, simulation, problem solution, and theorem proving. We shall distinguish between active and passive learning environments from a learner's activity point of view. During exploration through a passive learning environment a learner has only the possibility to read or watch information which had been prepared by the author in advance, for example to read texts from a database or to watch how a simulation program works for the author's data. The learner's activity is restricted in choosing a path through learning environment and his/her learning process has latent unobservable nature. During exploration of an active learning environment the learner's activity is directed to the construction of new information (at least it is new for him/her). For example, the student engages in construction of the structure and content of the subject being learned, creation simulation models for natural phenomena being studied, or devising a plan of attack for a theorem being proved.

Hypermedia information management systems present a new approach to electronic documents representation and processing (Smith & Weiss,1988). Now in the world a number of tools for creating hypermedia application have been developed - *Knowledge Management System (KMS), NoteCards, InterMedia, Document Examiner*, etc. Their application has demonstrated a great efficiency of information processing by users who are not specialists in computers (Akscyn and al., 1988). In education hypermedia-based learning environments can be used for learning and exploration of information models of the world. There are three types of hypermedia systems: those for knowledge presentation, knowledge representation, and knowledge construction (Nelson & Palumbo,1992). Knowledge presentation systems provide databases with associative links that can be browsed and/or searched in order to read or view information. Knowledge representation systems have means to make explicit (visual) the relations between portions of information. Such systems may use graphics for mapping the structure of a database. Knowledge construction systems provide users a means for creating information, allowing them to build and annotate nodes, create new and modify old links.

HELENA PROJECT: KNOWLEDGE REPRESENTATION

The *HELENA* (Hypermedia ELectronic ENcyclopedia Authoring tools) hypermedia project started in the middle of 1991. Its original goal was to make an authoring system for design and implementation of general purpose electronic encyclopedia environments (EEE). We had a background thought that hypermedia applications being developed with *HELENA* would be successfully used in education (Petrushin & Shevliakov, 1992). On the one hand the project is based on the traditional methodology for handbook creation. This includes the following stages: construction of a problem domain glossary, construction of an alphabetical glossary, preparation articles and illustrations for each term, and determination of links between terms. On the other hand *HELENA* was influenced by the experience of *ABC-1.5* authoring language development (Gritsenko, Dovgiallo, Petrushin, 1990). It borrowed from the language its frame-based knowledge representation of the problem domain.

The *HELENA* system consists of three programs: administrator, executor and constructor. The administrator program is intended for EEE users, registering and processing statistical data about their interactions with the EEE (which was not implemented in the first version). The executor program interprets EEEs for end users. The constructor program provides tools for EEE authors.

The *HELENA* authoring system implements the above methodology for handbook creation. Each term of the problem domain glossary corresponds to a concept of the problem domain. There are two types of links between concepts in *HELENA*: structured and associative. The problem domain glossary is presented graphically as a tree of problem domain concepts. Each node of the tree corresponds to a concept of the problem domain and a link between nodes C_1 and C_2 corresponds to relations like "C_1 is more general than C_2" or "C_2 is a part of C_1". These links we shall call *structured* links. An *associative* link is any non-structured link between concepts that is set by an author of an EEE.

In the stage of construction of the problem domain glossary *HELENA* visualizes the structure of the problem domain in the form of a tree. Here a user may edit the tree (insert/delete a node), edit names of concepts, visualize associative links of the current concept (overview), search a concept by name, and access to the information of the current concept (node) (Figure 1). The construction of the alphabetical glossary is made automatically. Every concept (node) is a frame that consists of slots. The slots can contain information of the following types: hypertext, picture, music, and executable program. Every slot is connected with a window which may be opened or closed. The designer and user may resize, move, open, close the window, and may extend it to the whole screen. Arbitrary intersections between windows are allowed (Figures 2-3). All operations with windows are under the control of the monitor of the original window system. Each slot of every frame has a modification bit. If the modification is allowed than the user may change the information of the slot while the EEE is interpreting. The determination of links between concepts is done using a corresponding text, picture, or music editor, where the user may set hypertext, hypergraphic or hypermusical links. All links connect concepts (i.e. frames, but not slots). The hypertext links are set in the traditional manner. To set a hypergraphic link an author draws a rectangle in the selected place of the picture and connects it to a concept name. Since the musical information is represented by text in *MML* (Macro Music Language), the setting of hypermusical link is similar to the setting of hypertext link. During the execution of a musical slot, the special window is put on the screen. It contains the picture of a loud-speaker, a small window with a textual label of the current hypermusical link. If the user "clicks" the

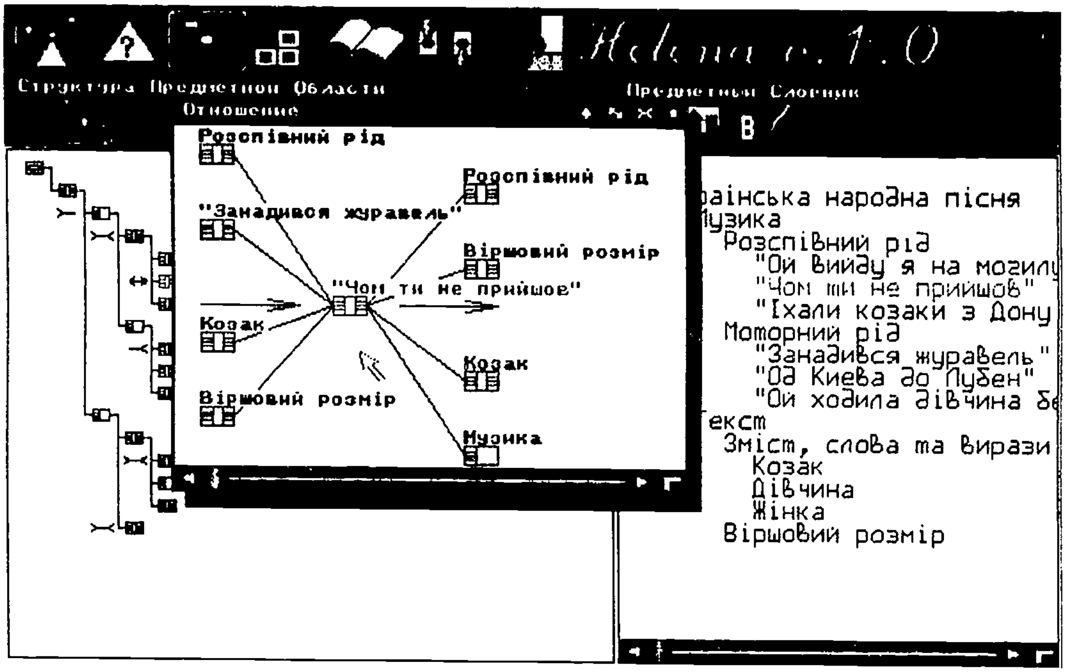


Figure 1. The main menu, the problem domain structure, alphabetical glossary, and local overview windows.

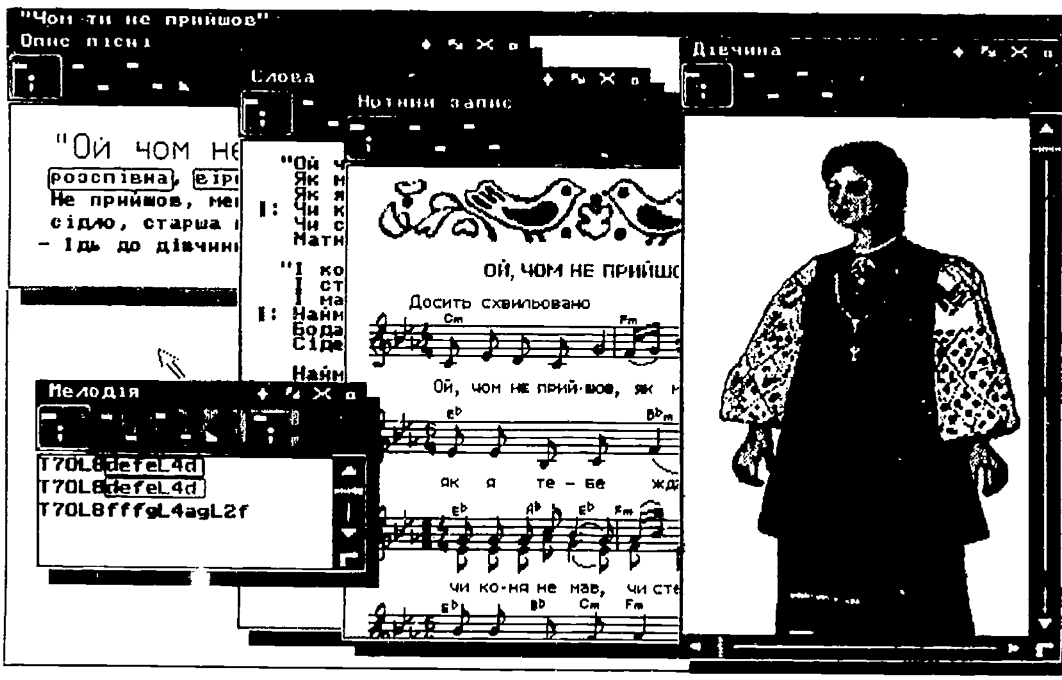



Figure 2. An example of frame structure.

BEST COPY AVAILABLE

left button of mouse, then he/she will see the concept which is associated with the current musical phrase (Figure 4).

Using *HELENA* we have designed and implemented two EEEs: *INFODENT* - a handbook for diseases of parodonteal tissue and associated stomatological terms and *UKRFOLK* - an Ukrainian folk songs encyclopedia. Some small applications were build to demonstrate the *HELENA* system's facilities, and for creation of conceptual design and quality evaluation systems for learning and tutoring programs of various types. The exploration of the EEEs have shown satisfactory results when they used by various audiences, and have revealed some weaknesses when they used by learners as learning environments.

The main weaknesses are:

- There is a lack of information about learner's activity. No learners' records (learner's path, using help and overviews) are kept. No information about visited nodes and slots like counter of visits or total visit time is collected. Hence, a teacher has no information for supporting decision making about learner's knowledge and the system has no information for supplying the learner by proper advice.
- There is unsatisfactory visualization of learner's activity during an EEE exploration. Visited nodes are not marked, and the learner's path is not available. This produces some navigation problems for learners.
- There are difficulties with the limited nature of overview facilities. The learner can see only direct neighbors of current concepts (nodes) and there is no way to visualize the "importance" of nodes.

HELENA PROJECT: KNOWLEDGE CONSTRUCTION

The most exciting application of the *HELENA* system was when it was used as a tool for creating hypermedia applications by secondary school teachers who had taken a course "Information Technologies in Education". The topic "Hypermadia Learning Environments" takes 12 hours which include a two-hour lecture about the hypermedia approach to information presentation and its applications, a two-hour class for demonstration and mastering *HELENA*, and an eight-hour workshop for creating a small hypermedia application. The workshop group was divided into small teams (2-3 person) and each team produced its own application having used the above methodology. The sources of information were journals, magazines, books from the library, and learner's individual knowledge. This was put into applications by using text and graphical editors, and a scanner for picture and text input. A teacher is afforded two possibilities for evaluating students' knowledge. One can

- watch the learner's application behaviour, and
- study the structure of the learner's application.

This explicit presentation of learners' activity results can help to evaluate the students' knowledge more deeply and correctly.

The experience with *HELENA* as an active learning environment has encourages us to modify the system. The new version of the system is under development now. It takes into account the above weaknesses and allows a learner to modify the structure of the problem domain. When an author creates a node he/she sets the node access mask which includes indicators which allow a learner to delete the node, add "a brother" or "a son" to the node,

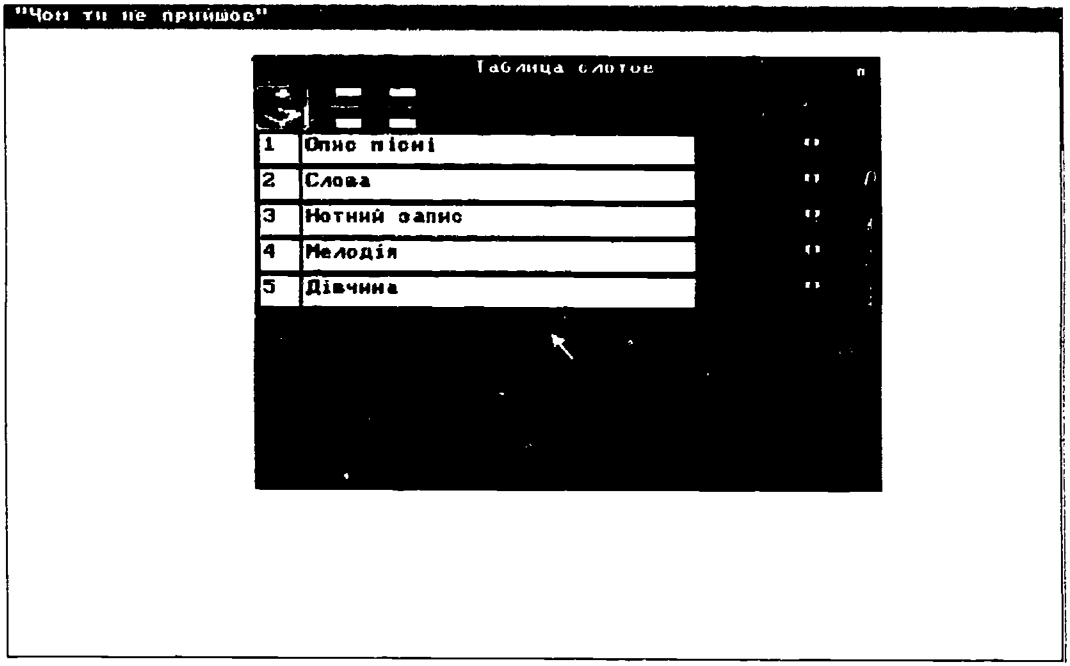


Figure 3. A table of slots.

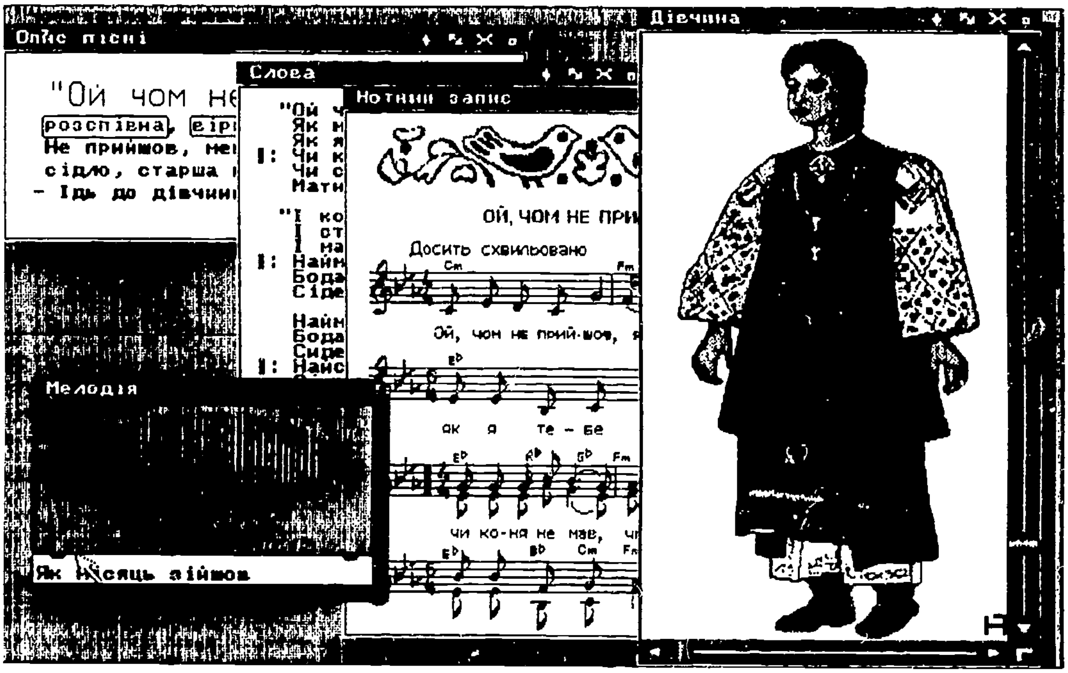



Figure 4. The interpretation mode: hypermusical links.

BEST COPY AVAILABLE

modify the node name, modify the node slots structure, delete a slot. We hope these facilities can improve the efficiency of the *HELENA* system as an active learning environment and a knowledge acquisition tool. Of course, this question needs to be experimentally studied.

CONCLUSION

The *HELENA* project began as a general purpose authoring tool for hypermedia applications, using visual facilities for knowledge representation. We hoped we could use it effectively for learning. But during the system exploration we have discovered that it had many weaknesses and the most effective way was to use the system as a knowledge construction tool. This experience will be take into account for the development of the next version. The further development of the authoring system is proposed to add new types of slots (for example, spreadsheet, video information, speech) and to improve interfaces both for authors and users. In additon we plan to conduct the wide experiment in classrooms.

We also have several open questions for future research and development. They concern learner modelling and knowledge representation problems and visualization techniques. Some of them are listed below.

How can the knowledge about instructional objectives help to solve learner's navigation problems? How should this knowledge be represented?

What types of information (slots) and links need to be added? How can we use 3-dimensional visualization techniques for concept mapping? How can overviews be controlled in an efficient way using individual learners' data?

How can hypermedia tools be combined with other knowledge acquisition techniques (checklist, repertoire grids etc.)?

We hope the investigation the above questions could improve effectiveness of application hypermedia tools for creative learning.

REFERENCES

Akscyn R.M., McCraken D.L., Yoder E.A. (1988) KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations. *Communications of the ACM*, *31* (7), 820-835.

Gritsenko V.I., Dovgiallo A.M., Petrushin V.A.(1990) Expert-Tutoring Systems: architecture and development tools. Proc. CEPES-UNESCO Int. Symp. on Artificial Intelligence in Higher Education, Prague, CSFR, October 23-25,1989. *Lecture Notes in Artificial Intelligence*, *451* ,46-53.

Nelson W.A., Palumbo D.B.(1992) Learning, Instruction, and Hypermedia. *Journal of Educational Multimedia and Hypermedia*, *1* (3), 287-299.

Petrushin V.A., Shevliakov A.S.(1992) An Authoring System for Electronic Encyclopedia Implementation. *Proc. East-West Conf. on Emerging Computer Technologies in Education*, Moscow, April 6-9, 1992, 255-257.

Smith J.B., Weiss S.F. (1988) Hypertext. *Communications of the ACM*, *31* (7), 816-819.

ERIC

Designing Multimedia Environments for Thinking Skill Practice

CLARK N. QUINN
MATT BOESEN
DANA KEDZIAR
DAN KELMENSON
ROB MOSER
School of Computer Science and Engineering
University of New South Wales
NSW AUSTRALIA 2033

To promote the development of thinking skills, an intrinsically motivating environment that naturally incorporates opportunities for practice of such skills is desirable. Computer adventure games are introduced as an environment that is motivating and requires cognitive problem-solving skills. Issues in the design of such games include instructional content, entertainment, and interface design. Drawing on our experience developing several such environments, we discuss these issues.

An important area of instruction is thinking skill instruction (Resnick, 1987) and an obvious approach is for computer support for such instruction (Quinn, 1988). While one approach would be to have the computer present information on cognitive skills (Quinn, 1992b), or to model the information processing aspects of such skills (Quinn, 1992c), the development of thinking skills not only requires the presentation of information on such skills, but on opportunities to practice and refine such skills (Collins, Brown, & Newman, 1989). Thus, a third possibility is for the computer to serve as a practice environment (Sherwood, 1990).

While there are numerous possibilities, one solution is to attempt to use the motivating nature of comptuer games (Malone, 1981). Several forms of games are available, with differing requirements for cognitive skills (Greenfield, 1984). One form of game that provides opportunities for problem solving behavior is the adventure game (Quinn, 1991a; Sherwood, 1991;

Adventure games as problem-solving

Adventure games naturally incorporate problem-solving skills (Grabe & Dosmann, 1988). In adventure games, the player takes on the role of a character in an imaginary world. The player directs the character, including the character's movement and actions in the environment. The character can hold and use some number of objects. Characters are subject to the physical characteristics of the world that they are in, so that while they can act in the world, the world can act on them. Included in this world are obstacles to movement that must be overcome to permit further exploration. The problem that players must solve involves the discovery of the specific actions, with the particular object, that must be performed at a particular location if the player is to explore any further. These characteristics - a motivating situation with embedded problems - provide the requisite research environment.

Adventure games usually incorporate a theme that motivates the exploration of the environment and provides the contextual clues that guide the problem solving. For instance, a game set in medieval times might have a quest for a religious artifact serve as the motivation for exploring the available world, and the obstacles could be either real hindrances such as brigands and foreign warriors or fantastic opponents such as dragons or trolls. The environment to be explored usually has a finite size and maintains a constant scale, but within these constraints it can range from a single house or dungeon to an entire continent or world. As mentioned, parts of the environment are not initially available for exploration and must be revealed by the correct use of an appropriate object at an appropriate location. For example, a portion of the world, including the goal, might lie across a chasm that cannot be crossed until a wand is found, brought to the chasm, and waved to create a bridge.

Objects to be found and used can be anything that can be logically or mythologically construed to have an effect on or in the world. The correct use of an object can usually be inferred from the properties of the object, as in the case of a weapon; the role that the object plays in the environment, as in the case of a magic scroll; or information provided during the course of the game. However, the use of the incorrect object or the use of an object in the wrong place can often lead to undesirable consequences, such as death for the character. Such a situation is illustrated in Figure 1, where the player must decide which object to use to overcome or pacify the spirit.

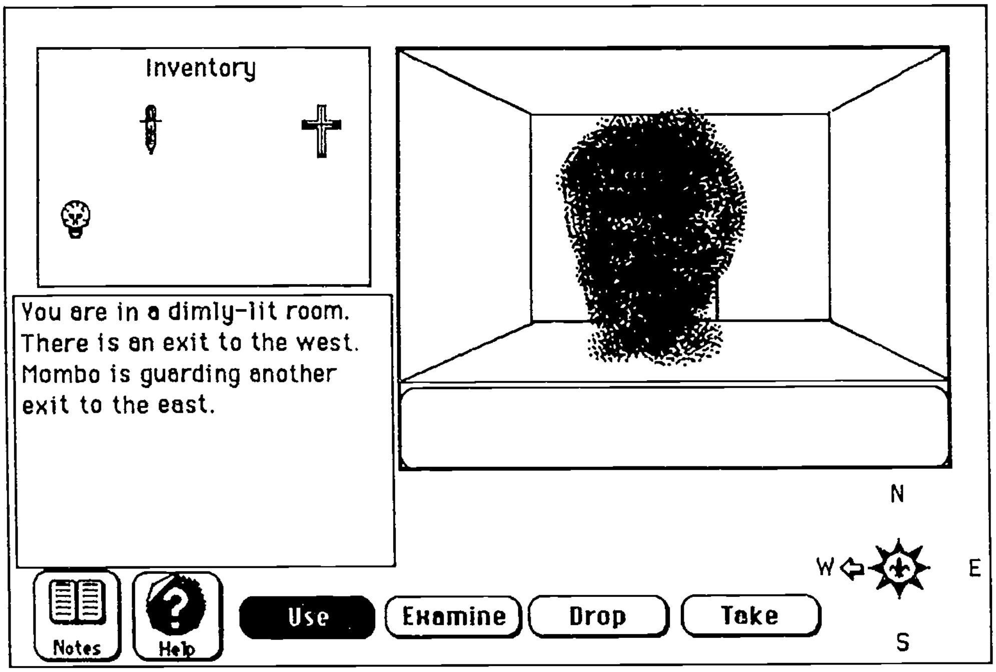


Figure 1. *Voodoo Adventure* screen

The games are made more difficult by the use of a variety of methods for obscuring the solution. These form miniature problems within the overall structure of resolving the game. Such methods include making the locations in one area difficult to distinguish, requiring the player to map the region, or making the discovery of the objects and their purpose less than obvious. Hints about the correct use of objects can be inserted into the course of the game by means of messages or clues. A further method is to allow more than one solution to a problem, so that if the wrong solution is used at one place, the solution object will not be available later when it is appropriate. In standard problem-solving parlance, the different locations in the environment correspond to the states in a state-space search, and movement between the locations comprises the operators that move between the states. However, certain spaces require another form of operator, that of using objects in combination with movement, to make further states of the space available for search. Of course, systematic trial and error can always, eventually, lead to the correct solution.

Design Issues

Inherent in this description of adventure games are several problems for design. A good game must have a coherent theme and must naturally incorporate specific problem-solving situations. Commercially available games are oriented for enjoyment, and typically do not focus on particular instructional skills. Other games follow the design principles described here. Additionally, the mechanisms by which students indicate their desired actions must be adequately adapted to the capabilities of the target audience. Finally, the realisation of such an environment must address the

BEST COPY AVAILABLE

practical issue of implementation. These problems cross boundaries of culture, cognition, instruction, and technology. Of course, only suggestions can be made in the first two cases, but the latter issues will be discussed in terms of two environments we have developed.

The first problem is also the most difficult. A requirement for creating a motivating environment is creating a coherent theme in which to embed the problems so that they are intrinsic to the activity (Brown, Collins, & Duguid, 1989; Cognition and Technology Group at Vanderbilt, 1990). A constraint on the problems is that they must be structured to reflect the desired cognitive property without violating the theme of the story. Existing situations or established myths may be used as themes, but there is no guarantee that specific problem-solving opportunities exist. It is difficult to find problems with the requisite characteristics in the obstacles faced by characters in existing fable, and it is unacceptable to creatively alter existing myth. One adventure game relied on the well-known Greek myths to establish a story line, mimicing the voyage of the Argonauts, but violated established knowledge in those myths. This game would both confound users who knew the myths, and would also misinform players who had not encountered the myths. It is also difficult to construct problems that contain the desired structure and are also believable situations, given the player's knowledge of the world.

It is more desirable to tap into the realm of fiction and popularly recognized alternative situations that are both commonly represented, frequently manipulated, and easily manipulable. Inspiration can also come from other forms of media including books, movies, and existing games. Obviously, science-fiction and fantasy are imaginative realms that are easy to manipulate in believable ways, but may have limited appeal. An alternative approach is to take some familiar situation and move it to an altered location. For example, a mystery could be located from a particular city to a generic city, combining aspects of many others. This abstraction allows the author to alter particular facts about the city to suit the nature of the cognitive skill, but still allows the student to draw on knowledge of how to act in a city to maximize the relevance of the task and to minimize interference from lack of contexutal clues. Such an approach is a common literary technique. Similiarly, history can be used to specify a time where enough detail is not available to make the situation unlikely, but the setting provides some structure to support student inferences of appropriate action. Each of these alternatives may have additional benefits. Science-fiction settings allow a natural introduction of technical and scientific information, historical settings can be instructional in the theme as well as the problems specificied, modern mysteries may highlight social ills as well. If the problems faced are, in fact, a replication of some real event the game instead becomes a simulation but may have instructional benefit as well. The overall approach is to pick a setting that naturally incorporates the thinking skill, and pick a thematic path that embeds practice opportunities.

A second problem is related to the first. An issue in the practical implementation of an adventure game is the particular cognitive skill to be emphasized in the problem-solving situation. There are a variety of solution strategies around which a scenario could be constructed. The information to solve the problem could be in a form that is amenable to a variety of strategies, or to a combination of strategies, to solve. For example, the information might be encoded, or it might require induction. The optimal solution strategy might be somewhat difficult, allowing for the expression and observation of alternate strategies. Often, it may be desirable to provide additional information for the solving of puzzles. If the skill is to use cognitive skills such as induction, information can be provided as clues in the game or notes that are available before the game. Additional materials can be external to the computerised environment, such as books, maps, or other learning aids. The game environment can be structured to depend on such materials, as is done in the popular Carmen San Diego games.

Careful use of alternate objects in the game can be used for instructional diagnosis. In the design of multiple choice questions, choices that would make sense if the student was applying an inappropriate solution, as in VanLehn's (1990) "mind bugs", can be used as diagnostics alternatives to the correct answer. For interface design, the same consideratins suggest the construction of alternative objects. For example, if a student were supposed to be choosing an object based upon structural similarity, a distracting object that had a similar function or name would indicate that the student was not focussing on the correct feature of the situation.

Another design issue in the construction of a computer-based game environment is the interface through which the player interacts with the program. The interface should support the activity without interfering in the execution. It also should reflect the way the user thinks about the task (Norman, 1986). A variety of interfaces have been used with computer adventure games, spanning the possibilities from the linguistic, with examples from simple two-word commands (use knife) to sophisticated language parsers (pull the sword from the stone) to the graphic, using the latest in input devices. For the purposes of assessing a cognitive skill, the interface has to be simple enough to be learned quickly, comprehensive enough to permit the full range of reasonable actions, and generic enough to forestall any hint of the solution to the problems.

Historically, adventure games developed in the same way that computer operating systems did, originating with text-based interfaces (indeed, some adherents prefer such interfaces). However, these interfaces require keyboard skills. As an alternative, graphic interfaces have developed for adventure games as they have for operating systems (Quinn, 1992d). The underlying concept is "direct manipulation" (Shneiderman, 1983; Hutchins, Hollan, & Norman, 1986; Ziegler & Fähnrich, 1988), that the user operates directly on graphically depicted interface objects. Such designs, to the extent that they support the appropriate inferences as to interface operation (Quinn, 1992a), improve system usability. Also, the graphic appeal of computer and video games would have been ignored, minimizing the potential attraction and consequent motivation for the players.

Once a graphic interface has been specified, the next task is to design the appearance of the screen. To do this, however, the semantics of the interface must be designed. Two aspects of the interface need to be determined: the actions that the player can instruct the character to perform and the feedback that the system gives to the player about the results of that action. The concern is to "design the action" (Laurel, 1991), drawing on the dramatic tradition.

The actions that the player can choose between needs to be limited to a representable set. In the case of *Voodoo Adventure* (Quinn, 1991a), a minimum set of actions was determined by the actions that a player must have available. A character needs to be able to pick up, or "take", an object at the character's current location, adding that object to the inventory of the objects that the character possesses. A character must also be able to "drop" an object at the current location. Due to potential ambiguities in the representation of an object, or the desire to initially obscure the potential application of the object, the player also needs to be able to have the character "examine" the object to determine its exact specification. Finally, the character needs to be able to employ an object according to its nature. Rather than allow a variety of different uses, a generic "use" option was made available. Although this limits the creative uses of an object, it also simplifies the interface and minimizes confusions for the player.

A different set of actions were used in the design of *Quest for Independence*, a game to support the acquisition of living skills. Rather than make all actions available at all times, certain actions were presented in certain circumstances, and different actions were available in others. In Figure 2, the actions available are navigation through the city, while in Figure 3 the player has entered a building and different actions are available. These actions are labeled and graphically represented to minimize requirements for language facility.

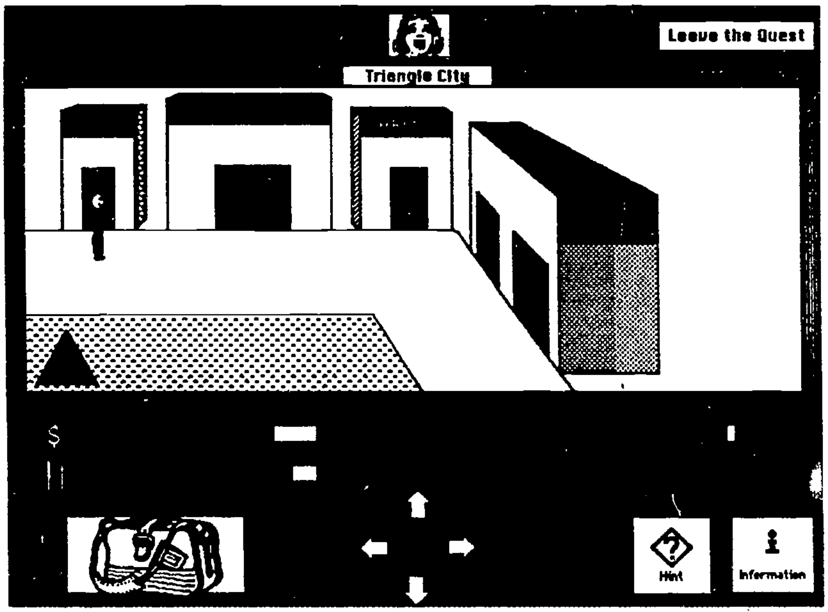


Figure 2. *Quest for Indepence*, in Triangle City.

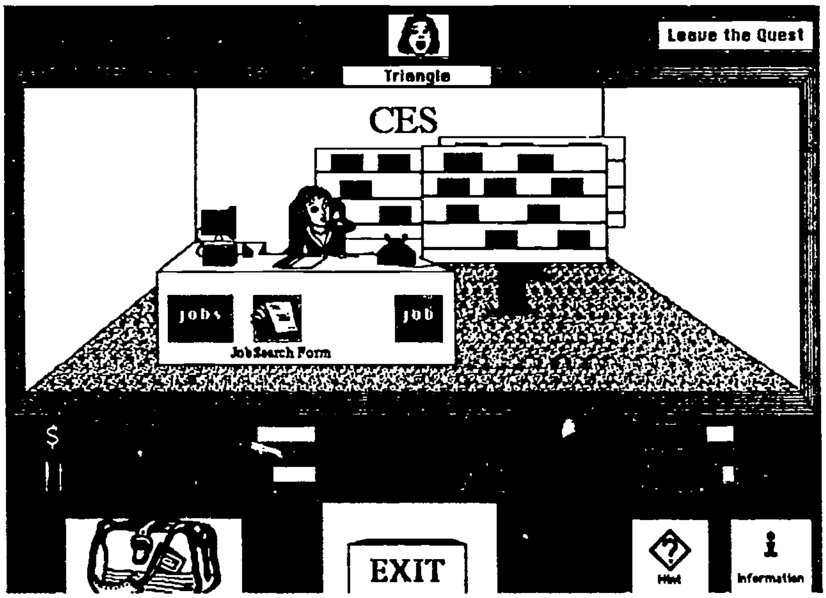


Figure 3. *Quest for Indepence*, in the CES.

Direct manipulation would suggest the representation of rooms, objects, obstacles, and inventory. Action would incorporate operation on these objects to accomplish a particular task. As the interface devices such as the mouse (or functionally equivalent pointing devices) become ubiquitous, a particular vocabulary has arisen to describe such direct manipulations. Using a pointing device in conjunction with a button, a user can select an object by moving a screen-based cursor over the represented object and pressing the button (clicking). Objects can be moved by selecting and then moving the cursor to the new location with the button held down (dragging). Thus, to "take" an object one would click and drag the object from the room to the inventory. Similarly, to "drop" an object would involve a similar click and drag from the inventory to the room. However, "examine" and "use" both involve actions on objects in the inventory. A popular mechanism is to double-select an object by a rapid sequence of two clicks at an object. It is not clear, however, as to which of these actions a double click should represent? And what direct action would support the other? One option is to use the double-click on an object for a description, which can include the room or the obstacle in the room, and dragging an object to the obstacle for use. This implies that there must be a clear delineation between the place in the room where objects appear and where an obstacle can appear. This also supports discrimination between scene-setting detail and game-playing items.

One problem that is particularly difficult for instructional environments is that the interface may interfere with the task. In most computer applications, the user is believed to be familiar with the task to be performed and can even use that information to comprehend the interface. In the case of instructional interfaces the task is learning new material, concepts or skills, and learning an interface can be an additional load. Further, computer skills cannot be assumed. One assumption in the direct-manipulation approach is that the user will be familiar with the necessary actions and will know about acting in an interface, but this assumption may not be valid in instructional environments. Alternatively, providing labelled buttons to represent actions can introduce a combination of object specification and action specification (in either order). The provision of both direct-manipulation and labelled buttons covers both issues as an interim solution. One problem is that labels will be dependent on language skills, and it may be that graphical depictions of actions may prove superior, but icon design is difficult, and no clear graphic representations for the actions of drop, take, use, and examine have yet been conceived.

A second issue is for the movement of the character through the environment. Navigation is a deliberate problem in some adventure games, but even unintentionally, the problems of remembering paths and determining movement options can be substantial. For example, Quinn (1991a) found that generic room depictions supplemented by individual textual descriptions were inadequate for some subjects. Movement problems can be confounded by creating curvilinear or non-planar paths. These might be deliberately exploited to focus on particular skills such as mapping, but careful control must be maintained. In other instances, navigational difficulties could interfere with practice of a desired cognitive skill. However, a clear and consistent set of navigation options must be made available to the player. The possible exits (unless deliberately impeded by an obstacle or strategically hidden) should be clear. Either a setting dependent navigation scheme can be used, where the particular exits of the current location are depicted, or a global navigation scheme can be implemented, such as movement in the four directions of the compass.

The complete interface needs to represent the current location in the game, the objects that the player has already obtained, any objects available at the current location, the available actions that can be performed, the directions in which the user can move to get to other locations, and feedback concerning the results of actions. To implement this graphically, there must be a picture of the room, pictures of the currently held objects (the inventory), pictures of the available objects, a representation of the possible directions to move, and a place to represent relevant feedback. The available screen is divided up into regions to accommodate these needs.

Once the interface has been designed, the next issue is how to implement the game. To completely implement an interface for a game is a daunting task; placing the burden of task management on the computer requires a sophisticated program. Various levels of support for interface construction exist, with a tradeoff between the difficulty of implementing the program, the flexibility of the resulting application, the distributability of the resulting application, and the modifiability of the resulting application. The possibilities range from using a programming language to the use of a canned application that generates adventure games. The available adventure game implementation packages had a number of problems, ranging from a text interface to lack of control of the output. Other solutions suffered from a variety of problems, including difficulty of implementation, freedom of distribution, or flexibility in use.

One solution, HyperCard, presents a simple yet robust environment for creating interfaces and provides a small but reasonable set of interaction possibilities. HyperCard is an implementation application that starts with the metaphor of a stack of cards and expands it with buttons that can be pressed to cause a variety of things to happen. A simple programming language, HyperTalk, allows fairly sophisticated semantics to be assigned to the buttons. HyperCard is readily available on the Apple Macintosh computer. HyperCard allows the designer to work at the conceptual level of interface objects, and support easy modification. Additionally, numerous extensions allow sound, animation, and video to be incorporated at a similarly high-level.

Conclusion

With HyperCard we have developed the two learning environments shown here. Several things are still missing. The implementation of the direct-manipulation version of the *Voodoo Adventure* game is in progress. Little use has been made of sound in that game, although some audio signals have been used in the *Quest for Independence*. The same situation holds for the use of animation. Other possibilities include expanding to include the use of video to support navigation.

An additional area for extension is the instructional component of the environment. To the extent that the cognitive strategy of the student can be ascertained by the use of distractor objects, remediation can be built into the environment. Other work on intelligent learning environments (e.g. Shute & Bonar, 1988) suggests that tutoring of cognitive skills can be realised (Quinn, 1991b).

We are also investigating other environments for development. Current projects are in progress with Toolbook and Visual Basic. One study that is just beginning is an investigation of the cognitive factors in development environments. We are also investigating other applications of multimedia for education, games with autonomous agents and simulations.

References

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher,* 18(1), 32-41.

Cognition and Technology Group at Vanderbilt, The (1990). Anchored instruction and its relationship to situated cognition. *Educational Researcher 19(6),* 2-10.

Collins, A., Brown, J. S., & Newman, S. (1989). Cognitive apprenticeship: Teaching the craft of reading, writing, and mathematics. In L. B. Resnick (Ed.) *Knowing, learning and instruction: Essays in honor of Robert Glaser.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Grabe, M., & Dossman, M. (1988). The potential of adventure games for the development of reading and study skills. *Journal of Computer-Based Instruction,* 15(2), 72-77.

Greenfield, P. M. (1984). *Mind and Media: The Effects of Television, Video Games, and Computers.* Cambridge, MA: Harvard University Press

Hutchins, E. L., Hollan, J. D., & Norman, D. A. (1986). Direct manipulation interfaces. In S. W. Draper & D. A. Norman (Eds.) *User Centered System Design: New Perspectives on Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Laurel, B. (1991). *Computers as Theatre.* Reading, MA: Addison-Wesley.

Malone, T. W. (1981). Towards a theory of intrinsically motivating instruction. *Cognitive Science, 5,* 333-370.

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.

Quinn, C. N. (1988). Pedagogical Framing and Cognitive Skills. *Technology and Learning,* 2(1), 6-7.

Quinn, C. N. (1991a). Computerized Cognitive Research: A HyperCard Adventure Game. *Behavior Research Methods, Instruments, & Computers,* 23(2), 237-246.

Quinn, C. N. (1991b). A Cognitive Skill Tutoring Environment. *Proceedings of the 1991 Conference on Intelligent Computer-Aided Training.* Houston, TX: NASA.

Quinn, C. N. (1992a). Explorability: Inferences at the Interface. *Proceedings of the Twenty-fifth Annual Hawaii International Conference on System Sciences.* Los Alamitos, CA: IEEE Press.

Quinn, C. N. (1992b, April). Teaching Cognitive Skills with Computers. *Proceedings of the East-West Conference on Emerging Computer Technologies in Education.* Moscow, Russia: ICSTI.

Quinn, C. N. (1992c, July). Cognitive skills and computers: "Framing" the Link. *5th International Conference on Thinking.* Townsville, Australia.

Quinn, C. N. (1992d). Hardware: Screen Design. In G.G. Bitter (Ed.) *Macmillan Encyclopedia of Computers.* New York: Macmillan Publishing Company. 442-450.

Resnick, L. B. (1987). *Education and Learning to Think.* Washington, D.C.: National Academy Press.

Sherwood, C. (1990). Adventure games in the classroom: a far cry from A says apple. *Computers Education*, 17(4), 309-315.

Sherwood, C. (1991). Computers and higher order thinking skills. In A. McDougall & D. Dowling (Eds). *Computers in Education.* Amsterdam: Elsevier Science.

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *IEEE Computer*, 16, 57-69.

Shute, V. & Bonar, J. (1986). Intelligent tutoring systems for scientific inquiry skills. *Proceedings of the Eighth Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

VanLehn, K. (1990). *Mind Bugs: Origins of Procedural Misconception.* Cambridge, MA: MIT Press.

Ziegler, J. E., & Fähnrich, K.-P. (1988). Direct Manipulation. In M. Helander (Ed.) *Handbook of Human-Computer Interaction.* Amsterdam: Elsevier Science.

PreScripts: A Typing Approach to the Construction and Traversal of Hypertexts

MARTIN RICHARTZ and MAX MÜHLHÄUSER
University of Karlsruhe, Telecooperation Group
P.O.B. 6980, 7500 Karlsruhe, Germany; [+49] (721) 608-4790

1 Introduction

A Hypertext can be viewed as a meaningful collection of information chunks, called nodes, arbitrarily interconnected by references, called links, which express semantic relations between nodes. This intuitively simple concept, the similarity to well-known concepts like 'book' or 'card-filer', and the easy-to-use direct-manipulation interfaces make hypertext popular with authors and readers of computer-based information.

Hypertext has a lot of potential. Viewed as the paradigm for structuring and handling computer-based information, it might soon be used in high-volume, multi-user, distributed, and mission-critical applications. But such use raises crucial questions. How much simplicity and ease-of-use can we preserve? Which concepts do we have to add?

In this context, we point out the importance of 'typing' (in the broadest sense) for many areas of computer science. Examples include macros in assembly languages, basic and structured data types in programming languages, class hierarchies in object-orientated systems, schema definitions in databases. We propose the development of a *typing concept* for hypertext and, in the remainder of this paper, will present our own approach to typed hypertext. We believe that by adding a powerful typing concept, the potentials of hypertext can be much better exploited; the challenge will be to maintain simplicity and ease-of-use as far as possible.

What does typing mean in the context of hypertext? We will distinguish below between *typing of hypertext elements* (nodes, links, ...) and *typing of hypertext networks*. While both are important, the former is comparatively well understood. The latter will therefore be discussed in more detail and is the main concern of our *PreScript* concept.

2 Typing of Hypertext Elements

In this section, we discuss typing of hypertext elements, to the extent necessary for the understanding of PreScripts.

Nodes: The primary hypertext elements are nodes and links. Simple Hypertext systems know only one fixed type of node (e.g., the 'card'), others allow all sorts of nodes but do not conceptualize 'types of nodes'. Among the systems that *do* provide typing for nodes, we advocate those which adopt the object-oriented typing approach to hypertext nodes. This can be done, e.g., by providing a base class 'hypertext_node' with operations such as 'edit', 'show', and 'show-by-anchor' (the object-oriented term 'method' will be used in the following instead of operation). This allows for easy introduction of new node and link types by sub-typing, proper handling of multimedia aspects by encapsulation, and unification of static and computed nodes (data and executable code).

Links: The adoption of the object-oriented approach leads to the most flexible concept for links, as well. A base class 'hypertext_link' is provided with subclasses 'static_link' and 'dynamic_link'. It is further enhanced by user-defined subtypes and thus offers full flexibility, an arbitrary numbers of pre-defined and well distinguished kinds of links, with minimal impact of new types to the underlying concepts and system.

Anchors: denote link sources or link destinations 'inside' a node. An open hypertext system with support for different media types usually supports external media editors (video editor, compound document editor, etc.), encapsulated via the methods of the object class 'hypertext_node'. It is up to those editors to define their view of an anchor, since they will usually have their own notion of an anchor (video_frame_number, document paragraph, image_cut_out, etc.). Therefore, an object-oriented *typing* approach seems to be appropriate for anchors as well.

Collections: there is little common understanding about 'collections' in hypertext. However, the increasing complexity of actual hypertext networks makes it very desirable to be able to define, e.g., *meta-nodes* (aggregates which represent a subnetwork) and *selections* (users' contexts such as versions, views, query results, and navigation paths).

3 Typing of Hypertext Networks: State-of-the-Art

Most of the candidates for the typing of hypertext networks were developed in the attempt to find 'formal models' for hypertext. Formal models may have different foci, such as: *compilation* (verification of desired properties, syntactic and semantic analysis, automatic 'translation', ...); *standardization* of a 'reference model' (to reduce ambiguities and to ease conformance testing); *openness* (the possibility for different hypertext systems to 'read in' the same formal model, allowing them to share hypertext documents); and *reuse* (instantiation of typed hypertext elements or network fragments for increased consistency and reduced authoring effort).

The hypertext standardization bodies (cf. Moline et al., 1990) were among the first to emphasize formal models and typing. Some authors applied existing or new formal specification approaches based on set theory (Halasz & Schwartz, 1990; Garg, 1988) or on predicate logic (Lange, 1990). Non-experts will usually find these very difficult to learn and use. This is in contrast to our basic requirement of ease-of-use.

For a more intuitive formal modeling approach, one can stick to the fact that hypertext networks (with static links) represent *directed graphs*. Many sciences (especially in popular science journals) tend to express complex abstract matters as directed graphs of one kind or another. Such graphs obviously imply low cognitive overhead and high information density. If we accept that even the 'card stacks and computed links' of HyperCard™ can be interpreted as directed graphs, then virtually all hypertext systems are largely based on the directed graph metaphor.

The only existing approach that the authors know of, which uses a similar metaphor, is called *Trellis*. Trellis (Stotts & Furuta, 1989; Furuta, 1989) maps hypertext networks and their browsing semantics onto a process model based on Petri nets. We believe that even Petri nets are too complicated for use by the average hypertext user, and that augmenting a graph by adding transitions and further nodes (as in Trellis) increases its complexity further. In our view, the most important feature of Trellis was described in (Stotts & Furuta 1990) at about the same time we decided to use *construction rules based on graph-grammars* in our PreScript approach (cf. Mühlhäuser, 1990; Mühlhäuser, 1992).

Just as programming language grammars formally define the syntactic rules for a language, graph-grammars define a 'language' for a usually infinite set ('family', 'class') of graphs. Recalling the fact that a hypertext network can be interpreted as a graph, this is exactly what we want: a way to express types of hypertext networks. Graph-grammars can be used to express repetitions (sequences 'node-link-node-link...', arbitrary numbers of links starting at the same node, etc.), hierarchies (several options for instantiating a 'non-terminal' or 'meta'-node), options (optional links and nodes), etc. Alas, as one can imagine, usual ways of depicting graph-grammars are again not very user-friendly. (Stotts & Furuta, 1990), for example, used a notation similar to the Backus-Naur-Form (BNF) for language grammars, which seems to be inappropriate for wide-spread use.

4 Requirements for PreScripts

In order to facilitate the work of hypertext authors and users by hypertext typing, we focus on the following requirements:

1) Navigation support and reuse of navigation strategies: we believe that simple pre-defined 'navigation paths' are not sufficient to prevent 'getting lost in hyperspace'; sophisticated guidance has to be given to the user on his way through a large hypertext network. For example, strategies and learner models for guiding a learner through instructional material have been adapted to hypertext-based material, where they figure as sophisticated (computed) navigation support. However, the application of instructional strategies in hypertext today is a disagreeable task: the author must understand the instructional theory and must implement the navigation support by adding pieces of code to each hypertext node. He must do this from scratch every time new instructional material is built.

A key goal with PreScripts was to be able to define such navigation strategies along with a *type* of hypertext network. Once a navigation-augmented type was defined, an author would be able to use that type when constructing or adapting a hypertext, thereby *automatically* 'applying' the navigation support that a navigation expert (e.g., instructional designer) had pre-programmed.

2) Construction support: for PreScripts, we build upon a type concept for hypertext *elements* as described in section 2. It will often be desirable to control how one is allowed to instantiate and combine different types of nodes and links in order to create a meaningful hypertext. For example, in a given subject domain, the types and construction rules may play the role of a schema definition in a database. For computer aided learning (CAL), one may define elements and construction principles for meta-models such as 'concept maps' or 'semantic networks'. PreScripts should support these kinds of construction rules.

3) Multi-layer support: hypertexts may be meaningfully split into several layers, each of which may take the form of a hypertext network in its own right: e.g., a 'concept map' in which instructional strategies are executed, a 'learner model graph', several networks of 'instructional transactions' (micro-strategies for teaching a single instructional unit), the 'domain knowledge network' which structures the learning domain independent of instructional aspects, etc. PreScripts must be able to support the construction of such multiple layers, their interrelations, and the simultaneous navigation of several networks, some of which may be visible to the user, while some may not be.

4) Manipulation support: hypertext authoring and reading cannot be totally separated: learners may make annotations, groups of authors may read and add information in parallel, etc. Therefore, PreScripts should support hypertexts whose construction 'never ends'.

5) Simplicity and ease-of-use: PreScripts should stay as simple and as intuitively usable as possible. E.g.,they should adhere to the 'directed graph' metaphor. But even with a graph-grammar based approach, user-friendly depiction and intuitive usability remains a challenge (cf. Trellis).

5 The Essence of PreScripts

PreScripts comprise three basic concepts: generic networks, rules, and procedures.

Generic Networks represent the most fundamental concept; this graph-grammar based part of PreScripts provides for the typing approach to networks of nodes and links. By means of a graph-grammar, a generic network defines a *type of hypertext network* by defining valid link types, node types and 'meta-node' types, along with '*graphical constructors*', i.e. a graphical depiction of how instances of the node and link types can be assembled into an instance of the network type which is

ERIC

defined by a generic network. Generic networks can themselves be viewed as a special kind of hypertext. Node and link types may comprise templates, i.e. default contents for the respective instances. Construction of hypertexts (instances of the type) is interactively supported by a syntax-directed graph editor.

Rules serve two major purposes: *additional construction support* (complementing generic networks, they provide consistency checks beyond the scope of generic networks at hypertext construction time) and *navigation support*. The latter feature responds to the first key requirement stated in section 4. As the user 'wanders' through the hypertext, the navigation rules are evaluated. resulting in several possible links (or a single one) for the user to follow next. By stating navigation support rules as part of a PreScript, authors or programmers can provide sophisticated reusable navigation strategies.

Procedures can be seen as an optional concept, to be used, e.g., by instructional design experts in order to provide user modeling. Procedures are invoked as the user traverses links or visits nodes. Providing a uniform coupling mechanism between the hypertext (the generic network, in fact) and the user model further enhances flexibility and reusability, in that different user models can be easily hooked into a hypertext, and a user model can be reused in different hypertexts.

5.1 Generic Networks

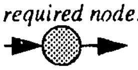
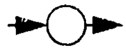
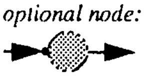
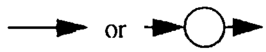
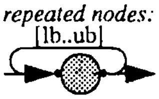
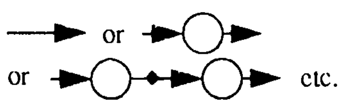
Basically, a generic network is a hypertext which defines mandatory, optional, repetitive, and alternative parts, as well as hierarchies. An instance of a hypertext is built by choosing among alternatives and options and by determining the concrete number of repetitions. We choose an intuitive graphical depiction for generic networks in order to reduce the 'formal mathematical flavor'; to this end, we also made a number of simplifications.

In the following, we distinguish between 'PreScript authors' and 'hypertext authors'; the latter construct hypertext networks as instances, i.e., under control of a PreScript generic network.

We use the term *expansion* if a hypertext author selects a node or link in a PreScript and instantiates it into the hypertext under construction. An important feature of PreScripts is the possibility to map a PreScript node or link onto an *existing* hypertext node. Thus a hypertext author can apply a PreScript 'a-posteriori' to an existing hypertext (representing, e.g., an existing knowledge domain), e.g., to make it compliant to certain instructional strategies. This a-posteriori technique is also denoted as *mapping*.

Below, we want to demonstrate the handling of *nodes* in a generic network first.

A PreScript author can constrain the type of hypertext nodes in a PreScript. If a hypertext author expands a node, he is offered the choice of valid types. In the process of mapping (expanding to an existing hypertext), the type information prevents mapping to an existing node which does not conform to the valid types. As mentioned, a template for the node contents may be provided by the PreScript author. Apart from the type constraints and templates, the role of a node within a generic network is determined through the use of the following *graphical constructors*:

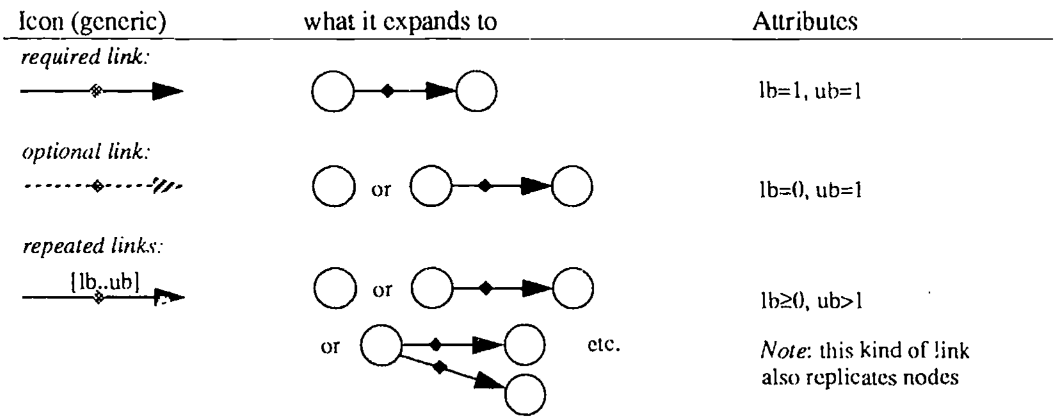
Icon (generic)	what it expands to	Attributes (cf. below)
required node: 		lb=1, ub=1
optional node: 		lb=0, ub=1
repeated nodes: 		lb≥0, ub>1

451

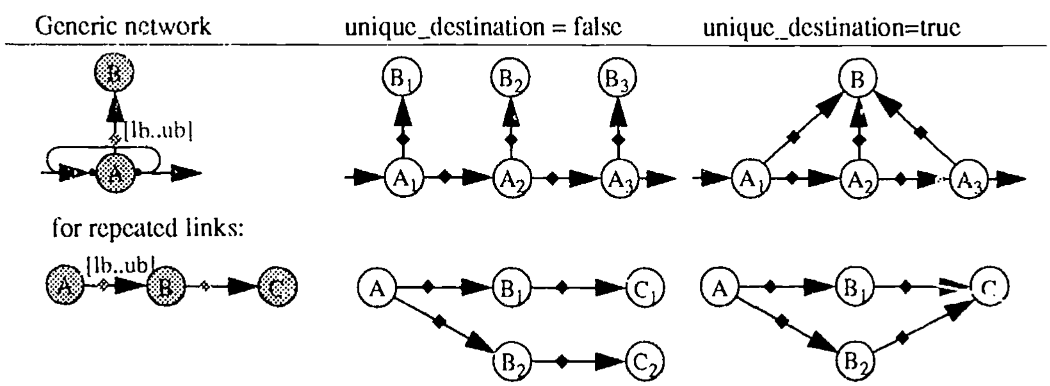
The attributes lb (lower bound) and ub (upper bound) specify bounds for the number of instances. From a formal perspective, all three graphical constructors could be reduced to a single one, deferring their distinction to the specification of appropriate lower and upper bounds; from a human interface perspective, however, this solution does not seem attractive.

Since a node type in a generic network can have many incoming and outgoing links, the optional and repeated nodes comprise a pair of distinguished incoming and outgoing links, indicated by the diamonds at their endpoints. If *no* node is instantiated at all (i.e., an optional node is not chosen or the number of repetitions equals zero), the two bulleted links are merged (all other links are removed). For *multiple* expansions of a repeated node, the bullets indicate three things: the link to the first node and from the last node in a sequence, and the links between subsequent nodes (cf. to the link type marked with [lb..ub]); all other links are replicated for each expanded node.

We now proceed to the handling of *links* in a generic network. Type constraints and templates for links can be provided just as for nodes. The graphical constructors for links are as follows:



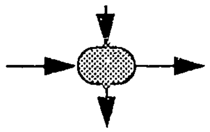
Links have another attribute called 'unique_destination' which is interpreted for the additional (without diamonds) links outbound to 'repeated nodes' or outbound to target nodes of 'repeated links'. The effect of this boolean attribute is described in the figure below:



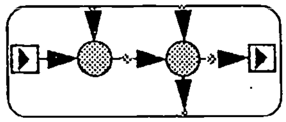
For 'repeated links', the unique_destination attribute can be thought of as the complement to the repeated link itself. While the latter 'splits' (forks) paths, a unique_destination 'merges' (joins) paths.

Support for modularity / hierarchy: The PreScript generic network mechanism supports *meta-nodes* as a means for modularity, i.e. hierarchical structuring. A meta-node expands to a subnetwork of its own; the way in which a meta-node may be expanded is defined by another generic network (or several alternative ones) which may or may not be defined within the same PreScript.

In the encompassing generic network, meta-nodes are depicted as rounded rectangles. The 'inside', i.e. the generic network for a meta-node, contains nodes and links *plus* mandatory inbound and outbound 'pseudo nodes' and optional 'endpoints'. By default, every link of the encompassing hypertext under construction which is inbound (outbound) to the meta-note is automatically connected to the inbound (outbound) pseudo node; at navigation time, pseudo nodes are invisible to hypertext users.



Generic meta-node



Generic network for meta-node

Generic network inside meta-node, including its two pseudo nodes and three endpoints

In section 2 we required 'selections' in addition to meta-nodes; at present PreScripts support these as 'navigation selections' only; versions and retrievals are not yet supported.

5.2 Rules and Navigation Support

As stated earlier, rules provide both additional construction support and navigation support. We want to concentrate on the latter here.

Navigation rules can be associated with links. They are evaluated as the user 'wanders' through a network. When the user 'arrives' at a given node, the rules of all outbound links are evaluated and yield an *importance* value for each link. Based on a threshold function and a display function, the hypertext browser determines which links are presented to the user for navigation. The navigation rules for a link can refer to methods (and, in part, rules) of adjacent nodes and links.

Usually, navigation rules are associated with the links (link types) of a PreScript by a PreScript author and are later 'automatically inherited' by an actual hypertext under construction. Thus, sophisticated navigation support can be provided by a PreScript author, e.g. an instructional design expert, so that hypertext authors need not be experts in this area. However, a hypertext author can always override navigation rules.

The navigation context (user model) must be maintained on a per-user basis. This leads us to three levels of specialization: PreScripts (common to a class of hypertexts), hypertexts, and the different navigation contexts of the users working on a hypertext (this corresponds to our requirement of multi-layer support). If we want to allow different users to *manipulate* a hypertext, we have to maintain (partial) copies of the hypertext and the corresponding PreScipt(s) for every user (cf. to our requirement of manipulation support). Different users may thus perceive quite different views of the same basic hypertext at navigation time.

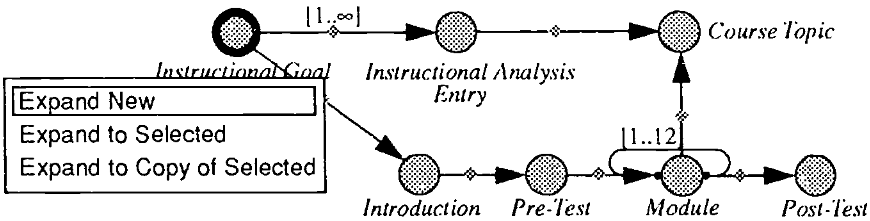
5.3 Procedures

Just as rules determine *conditions* under which a user may navigate along a certain path in the hypertext network, procedures determine *actions* to be carried out when navigation takes place. They are used to control node activation and to update the navigation context during navigation.

Like rules, procedures can be associated with the nodes and links of a generic network. They can be provided by PreScript authors along with the rules in order to implement sophisticated navigation support. While we intend to provide a graphical-tabular approach for the specification of procedures, the first version of PreScripts uses a traditional approach to procedures. In fact, procedures are defined as object-oriented methods: a link may include a *first-traversal* method which is invoked when a user follows this link for the first time and an *each-traversal* method which is invoked each time the user traverses the link. Accordingly, *first-visit* and *each-visit* methods exist for nodes; in addition, *callback* methods may be invoked from within an activated node. For the actual activation of a node, the activation procedure needs to call the default 'show' or 'show_by_-anchor' method of the node, cf. section 2.

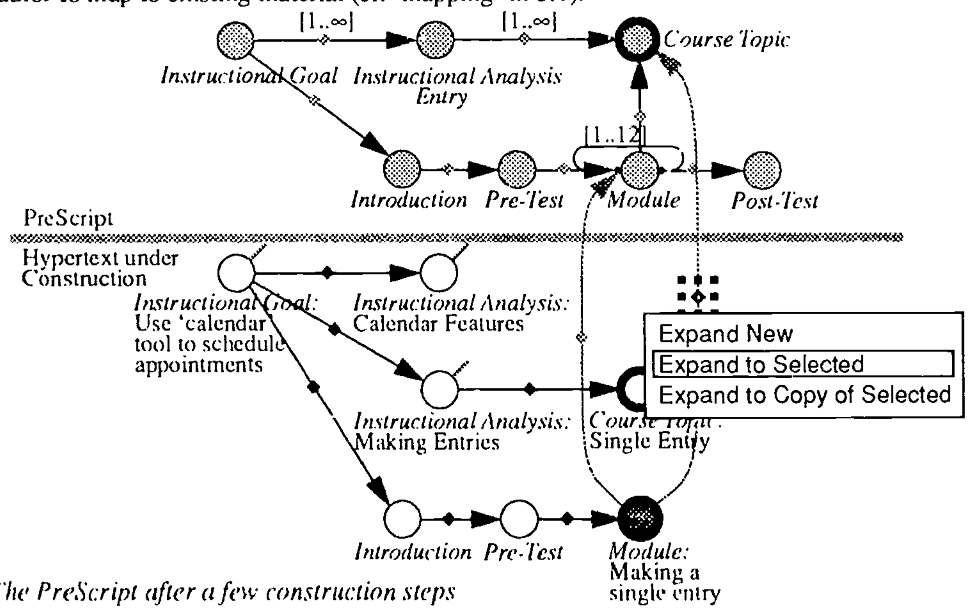
6 An Example PreScript

In order to demonstrate the use of the aforementioned concepts, we want to consider a simple example PreScript. We focus on the construction support given by the *generic network* part of a PreScript. The PreScript below represents the overall structure of a computer-based course (only a small section is presented here).



An Instructional Designer's PreScript at the beginning of course creation

This PreScript is interpreted as follows: after stating an overall instructional goal for a new computer-assisted course, an instructional analysis is conducted which details the subgoals that the students are to achieve (instructional analysis entries). In the next phase, course topics are derived from the instructional analysis entries. The lower part of the PreScript defines the course structure that is later accessible to the students (in contrast to the upper 'instructional analysis' part). The popup menu shown relates to the 'Instructional Goal' node selected; the author is about to select "Expand New" from the menu to instantiate a *new* node. The other alternatives would allow the author to map to existing material (cf. 'mapping' in 5.1).



The PreScript after a few construction steps

454

The above diagram shows the PreScript and the newly created hypertext several steps later. PreScript and hypertext are visually kept apart, although relationships between them are retained.

Note that care has been taken to keep the graphical view easy to perceive. E.g., links from the hypertext under construction to the PreScript are only fully drawn (in gray) for the node which is actually selected (here: node "Module: Making a single entry"). For all the others, only small gray tails are shown as visual clues (e.g. for the node labelled "Instructional Goal: Use 'calendar' ...", this clue indicates that more links to "Instructional Analysis Entry" nodes can be created, as the PreScript allows an unlimited number of these links). For the currently selected node, "Module: Making a single entry", the PreScript requires a link to a node of type "Course Topic". In our example, the author wants to connect this required link to the node "Course Topic: Single Entry". He has selected both the (still uninstantiated) required link and the hypertext node "Course Topic: Single Entry". The author can now establish the link by choosing "Expand to Selected" from the menu.
As one can see, an adequate user interface has to provide high flexibility to its users, e.g.: instant display or suppression (cf. gray 'clue') of incomplete links; different views of the hypertext and corresponding PreScript, cf. (Furnas, 1986); and consistent handling of multi-layer displays.

7 Conclusion

We have stressed the importance of typing for a more successful spread of the hypertext gospel into different fields of computer science, especially CAL. We have presented a hypertext typing approach called *PreScript*, that keeps the cognitive overhead for authors reasonably low and provides particular support for reusable navigation strategies and user models.

The proposed concept has been developed within the framework of the NESTOR project, a joint development effort for a distributed, computer based learning and authoring environment by the University of Karlsruhe, Digital Equipment's Karlsruhe based Campus Engineering Center, and the Heriot-Watt University at Edinburgh. The PreScript graphical editor and runtime support are currently in the process of implementation, preparing it for further evaluation later this year.

8 References

Furnas, G.W. (1986): Generalized Fisheye Views. *Proc. ACM SIGCHI '86*, 16–23.

Furuta, R. et al. (1989): A Spectrum of Automatic HT Constructions. *Hypermedia* **1**, 179–195.

Garg, J. (1988): Abstraction Mechanisms in Hypertext. *CACM* **31**, 862–871.

Halasz, F., Schwartz, M. (1990): The Dexter Hypertext Ref. Model. *(Moline et al., 1990)*, 95–133.

Lange, D.B. (1990): A Formal Model of Hypertext. *(Moline et al., 1990)*, 145–166.

Mühlhäuser, M. (1990): 'Intelligenz' vs. Hypermedia in Lehrer-/Lernerumgebungen. *Reuter, A.: GI – 20. Jahrestagung II, Springer Verlag IFB258*, 409-413 (in german).

Mühlhäuser, M. (1992): Hypermedia and Navigation as a Basis for Authoring / learning Environments. *Educational Multimedia and Hypermedia* **1**, 51–64.

Moline, J. et al. (1990): *NIST Hypertext Standardization WS, Gaithersburg, MD. NIST SP500–178.*

Stotts, P.D.; Furuta, R. (1989): Petri-Net-Based Hypertext: Document Structure with Browsing Semantics. *ACM TOIS* **7**, 3–29.

Stotts, P.D., Furuta, R. (1990): Hierarchy, Composition, Scripting Languages, and Translators for Structured Hypertext. *Proc. Echt '90, Versailles. Cambridge University Press*, 180–193.

ERIC

The "Virtual Language Lab": Remote Foreign-Language Learning Through Digitized Speech Technology

IAN M. RICHMOND
Department of French
The University of Western Ontario
London, Canada N6A 3K7

Preamble

In this paper, I shall discuss the feasibility of using digitized speech technology to create computer-assisted language learning (CALL) audio materials suitable for distribution through standard telephone lines and of developing the necessary delivery system. The proposed system is intended to be an integral part of a projected self-study, multimedia language learning center that will replace an aging language lab facility. Thus, the system discussed here is not intended as a total language learning solution, but as one element of a much broader system incorporating speech, video and more traditional CALL materials. The remote-access audio element is of particular interest, though, because it offers a relatively low-cost way of maximizing the physical learning center facility while enhancing the students' learning experience.

Methodological Considerations

The traditional audio language lab has been in general use for about the past thirty-five years. Because it has never quite lived up to the high expectations the foreign-language teaching profession had originally held out for it (Hutchinson; Politzer; Stack), however, for the last twenty years or more, its use has given rise to widespread disenchantment and dissatisfaction among both teachers and students (Cammish; Iandoli; Mueller *et al.;* Paramskas & Thomas).

This state of affairs is not altogether surprising. The language lab as we have known it for the past forty years was conceived in the days of the audio-lingual methodology (Claudel; Nimmons). As Politzer points out, it is ideally suited to the repetitive stimulus-response approach inherent in that methodology. As the audio-lingual method has been replaced by cognitive methodologies, with a greater emphasis on communication and comprehension, the audio-lingual lab and the stimulus-response drills which it does best have lost most of their methodological underpinning (Cammish; Nimmons). Despite this general disenchantment, however, sound reproduction technology can still be a valuable adjunct in the language learning process (Hammerly; Mueller *et al.*; Rivers), along with the video and computer technologies developed in recent years. The trick is to find more effective and flexible ways to use it than those associated with the traditional language lab.

One of the more interesting alternatives to the traditional, audio-lingual language lab is a self-study facility that permits students to work with audio materials at their own pace, in their own time and outside the language lab environment. Mueller *et al.* found that a tape-lending system meeting these criteria enhanced their students' learning experience: "... not only did those students in our experiment who worked outside the language laboratory spend more time on the tapes overall than those who worked in the lab, but the quality of the time spent was also higher" (589). My own institution has used a similar takeout system with great success since the late 1970s.

Particularly since the advent of the now ubiquitous "Walkman" portable tape player in the early 1980s, there is little justification for requiring students to conform to a laboratory timetable in order to listen to material they can play equally well on their own equipment. Students appreciate the flexibility that a self-study facility affords them: "The choice is now up to the students: they decide where they want to study, when they want to study, and how long they want to study. Moreover, since the daily schedules of most students are overcrowded, this freedom of choice is most welcome" (Mueller *et al.*, 589). Hammerly found that "Adult students in particular find self-instructional programs advantageous, as their schedules make class attendance difficult, they may live far from where the classes are held, and they like independence, preferring to be self-reliant rather than follow orders" (608). Since more and more university and college students choose or are forced to work at part-time jobs—about forty-five percent of the students at my own university work eleven or more hours a week—this sort of flexibility is becoming, not only desirable, but essential.

A self-study, external facility precludes, of course, the possibility of monitoring the students while they orally repeat or respond to recorded material. It also usually precludes the possibility of letting the students record and listen to themselves. Since these possibilities have been among the main justifications for the traditional audio-lingual language lab (Claudel; Hammerly; Nimmons; Rivers), giving them up represents a considerable departure from traditional language lab practice. Nevertheless, there is evidence that the recording and playback of student responses may not be of much benefit, and that it may well contribute to error fossilization (Mueller *et al.*; Nimmons). Mueller *et al.* also found that monitoring of students was largely ineffective. Thus, they conclude, "It seems, then, that self-correction and laboratory monitoring are mixed blessings at best. The most enthusiastic students may well want to take advantage of them, but even they may not profit much from them" (586-587). For his part, Hammerly (597-598) cites ten reasons why monitoring students in the language lab is not a worthwhile practice. Although Hammerly is in favor of recording student responses so that they may be compared with a model, he points out that recording and comparing is effective only if based on thorough training in phonetic discrimination and articulation. Hammerly acknowledges that such training is most often lacking or inadequate. Finally, the emphasis on comprehension (listening), as opposed to production, implied by an unmonitored, self-study audio program is well supported by current second-language acquisition theory (Courchêne *et al.*; Nimmons). It seems, then, that eliminating the ability to record student responses and especially to monitor students while they work will not likely prove detrimental to second/foreign language learning and may even be beneficial, especially in light of the increased motivation reported by Nimmons and by Mueller *et al.*

As the preceding discussion demonstrates, there appears to be considerable methodological support for the implementation of a self-study system in place of, or at least in tandem with, the traditional classroom-style language lab. Although the existing literature is limited to discussions of tape-based library systems (e.g. Nimmons, Mueller *et al.*), there is no reason why the concept cannot be expanded to include a digitized speech system implementing remote access to audio materials by telephone. The arguments in favor of a tape-lending, self-study facility, briefly presented above, apply equally to a phone-in system using digitized audio materials stored on a central server. The two systems are not mutually exclusive; they can, in fact, complement each other very nicely, and probably should so as not to overload a phone-in system. As I shall try to show, however, a phone-in, digital system has unique advantages that cannot be matched by any analog (or even digital) tape facility.

Technical Background

Digitized sound is quickly becoming a standard feature of microcomputers. The Apple Macintosh has long had digitized sound reproduction capabilities and, more recently, has acquired the system-level ability to record sound directly onto a computer storage medium. Thus, all newer Macintosh computers can record music, sound effects or speech directly to a diskette or a hard disk without the addition of extra hardware or software. The MS-DOS world has not been left behind in the area of digitized sound. MS-DOS compatible computers have acquired much the same sound recording and reproduction capabilities as the Macintoshes through the development of relatively inexpensive add-on sound boards, such as the *Sound Blaster*. At the same time, *MS Windows* 3.1 has incorporated a simple sound recorder very similar to that included with the Macintosh's System 7 operating system.

Users are not limited, however, to the fairly primitive recording and editing functions provided by the Macintosh's system software and by *MS Windows*. Users of either environment have access to a host of add-on recording and playback hardware, ranging from simple microphones to sophisticated MIDI music systems. Similarly, a wide variety of sound editing and sequencing software has been developed and is now readily available to meet virtually every need.

Digitized Sound vs. Analog Technology

The progress in this area has been such that we are now beginning to see computerized language laboratories and language learning centers using digitized sound recording and playback technology in preference to the traditional analog tape recorder. The Wuppertal Language Institute and the University College of North Wales are two institutions where digital, microcomputer-based technology has replaced the tape recorder in the language learning center. Although the existing heavy investment most institutions have in analog technology means that we will likely not see wholesale replacement of traditional equipment for several more years, it is inevitable that more and more institutions will opt to replace aging audio tape recorders with microcomputers equipped to record and play back digitized sound.

The main advantage of digitized sound over analog tape recording, the advantage from which most others derive, is that a digitized sound file can be treated like any other computer data file. This means, among other things, that the sound recording can be copied from one disk to another with no loss of sound quality, that it can be compressed for storage and that it can be transmitted from one place to another by modem or across a network. In particular, however, it means that a digital sound data file can be accessed in true random manner and with the customary speed of computer access. An analog tape recording, on the other hand, is rigidly sequential, requiring that much time be spent winding and rewinding tape in order to find a given item of data. This apparently simple distinction is of enormous significance for the development of computer-assisted language learning (CALL) software that needs to integrate sound with text, animation or real-time video.

Despite heroic efforts by developers and hardware manufacturers alike, integration of analog, tape-recorded sound with CALL software has never met with great success. The difficulty of locating discrete items of data on a sequential tape and that of designing a reasonably priced tape player that can position a tape head quickly enough and with sufficient precision to allow discrete sound items to be integrated with a computer program have conspired to render audio tape recording impractical for most CALL purposes.

Video-disc technology, of course, resolved all the above problems except that of cost. While video-disc is a marvelous and promising technology, the simple fact is that the cost of producing an educational video-disc is well beyond the means of all but the wealthiest institutions. Even the cost of purchasing ready-made interactive video-disc materials for CALL has severely restricted the

ERIC

dissemination and development of this technology. On the other hand, digitized sound technology is now within reach of all who can afford a decent microcomputer.

Remote Access Through Digitized Sound

The ability to integrate digitized sound files with various types of software opens up a number of pedagogically interesting perspectives that go far beyond the usual on-site applications integrating sound with text and/or video. One such perspective is the possibility of combining digitized sound with voice-mail and touch-tone response technologies to create interactive audio learning materials accessible in a remote mode to any student with a touch-tone telephone.

Voice-mail and touch-tone technologies are already well established in non-pedagogical domains. In the business world, for example, they are regularly used as automated answering services and ordering services. By combining digitized speech output, to provide directions, with the customer's touch-tone responses as input, it is possible to conduct quite complex business transactions without requiring the customer to interact with a human employee. Even universities have discovered this technology and, in fact, regularly use it for automating switchboards and for allowing students to register by telephone, "typing" their identification numbers and desired course and section designations on the telephone's keypad in response to digitized prompts and queries.

In systems of this type, the caller interacts with the system to input a response. The system can store the caller's response in a database for later use by a human worker or by another program or it can use the caller's response to branch to a different routine and initiate some new action to which the caller will be asked to respond. In this way, the caller can be led through a complex series of operations. In most cases, the caller can also interrupt the series at any time to end the interaction or to return to some earlier point in it. Of particular interest for pedagogical applications is the fact that the caller's responses can be used to change the database from which the voice-mail system fetches its prompts and questions. In this way, the voice-mail/touch-tone system can become the gateway to a variety of activities that can be launched under the caller's control. The impending implementation of the recently developed ISDN (Integrated Speech and Data Network) fiber-optic telephone systems by the North American communications companies will give further impetus to this technology, which already has enormous potential for distance education and for independent study by telephone.

Advantages of the "Virtual Language Lab"

By combining the digital sound capabilities of the microcomputer with voice-mail and touch-tone technology, it is possible to create a genuine electronic classroom "without walls." Without leaving his or her own home or residence room and with no equipment other than a touch-tone phone, the foreign-language student with access to such a system can enjoy many of the same benefits as students using an on-site electronic classroom or language learning center.

The application of this combined technology to distance-learning courses is obvious: being able to access interactive audio materials by telephone, gives students in such courses virtually the same advantages as students in regular language courses. Less obvious, perhaps, are the benefits to students who may find it physically difficult or impossible to get to or to use a normal language learning center. Hearing impaired students, for example, could use amplification devices that would be impractical in a multi-user language learning center because of their impact on other users. Shy, anxious students who are made severely uncomfortable in traditional classrooms and in language labs where their privacy is not assured could derive great benefit from such a system. Finally, all students can gain from the independent-study potential of this combined technology, which provides access to

ERIC

key audio language-learning materials twenty-four hours a day at the student's convenience, with immediate, although limited, feedback and with privacy as complete as the student wishes to make it.

Despite the initial cost of setting up a remote audio learning system, the application of digital sound to remote-access technology can prove very cost effective. Indeed, adoption of this technology amounts to greatly increasing the language learning center's capacity, or throughput, with no increase in its physical size or personnel needs. In this way, one can truly speak of a "virtual language lab," one that exists in time and in the mind of its user as it is being used, but that has no physical reality. Once the system is in place, the addition of a single telephone line can expand its capacity far beyond what would be possible by just adding another row of booths to a traditional language lab or a couple of workstations to a language learning center.

Technical Description

A number of problems are inherent in the design of a system for remote delivery of digitized audio materials by telephone. The main ones can be identified as inadequacy of disk storage space, incompatibility of digital sound files among various applications and the limited interaction allowed by touch-tone technology.

The first of these problems appears to be the easiest to solve. One simply has to have a big enough hard disk! That obvious solution is not perhaps sufficient, however. How big is big enough? Even a multigigabyte hard disk will eventually fill up with sound files, which occupy about 12K of disk space for each second of recorded sound, even at a fairly low sampling rate. This means that one minute of speech recorded with adequate, but not outstanding, quality occupies the equivalent of one 3.5 inch diskette. Two minutes of recorded sound will occupy the equivalent of a high density 3.5 inch diskette, or one twentieth of the space available on an otherwise unused 20 megabyte hard disk. A higher sound sampling rate for increased quality will require even more disk space. If we multiply this usage by the number of minutes required in a weekly lesson for a language course (probably about twelve minutes, since silence does not need to be stored) and then multiply that by the number of courses using the system, we find that our gigantic hard disk is hardly big enough, especially if the same disk is being used to store system and network software and the applications that create and use the sound files as well as whatever other applications the language learning center may use. Nonetheless, the solution to the storage problem can be solved by the use of removable storage media. Currently, the most cost-effective removable medium seems to be the Bernoulli system of removable disk cartridges. Each removable cartridge can store up to 150 megabytes of data and changing cartridges is a simple matter. A more expensive alternative is to use rewriteable optical or magneto-optical disks. The former can hold about one gigabyte of data, while the latter generally weigh in at around 120 megabytes. Either solution, however, requires that the system's primary hard disk be reserved only for materials in current use. The secondary storage medium, the removable one, is therefore used for long-term storage. This requires that some sort of cataloging system for the removable media be put in place so that the appropriate files can be loaded to the primary hard disk each week. This is no more complex, however, than ensuring that the appropriate tapes are available each week in a traditional language lab.

The problem of incompatibility of sound files among different applications is a somewhat thornier problem. Particularly in the MS-DOS environment, there is little compatibility among sound files produced by different applications. The files produced by a popular sound board such as the *Sound Blaster*, for example, are not readable by the readily available voice-mail systems, such as *Big Mouth* or *Watson*, and vice versa. Microsoft's adoption of the *Sound Blaster* format as the standard for MS Windows' multimedia extensions will no doubt bring about a considerable degree of standardization in this area, but for the moment it remains a problem.

4C.)

ERIC

The solution I have adopted to this problem is to opt for the Macintosh computing environment. Using the Macintosh for this sort of project has one main advantage: the digitized sound input and output are handled at the system level. For this reason, digitized sound files on the Macintosh, at least under the latest version of the operating system, are compatible with all applications using digitized sound. This means that sound files need be recorded and stored only once and yet can be accessible to the voice-mail software as well as to the software to be used on-site in the language learning center.

Finally, the problem of the limited interaction permitted by touch-tone technology must be addressed. The key here is to find a voice-mail system with as much flexibility as possible. The remote delivery system I am currently working on is based on a powerful, programmable voice-mail system, *PhonePro* by Cypress Technologies, which allows the caller to enter responses using any keys on the telephone's keypad and also to respond by voice and have the message recorded and saved in a database. It even provides for alphabetic input by using a two-key system, whereby the caller presses a key bearing an alphabetic character and then presses a numeric key to indicate which of the letters on the key the key press is intended to represent.

Certainly, none of these input options is ideal. Ideally, the caller would respond orally and have his or her oral response recognized by a sophisticated voice-recognition program in whatever foreign language he or she was using. Such an ideal system will require the development of voice-recognition software with capabilities that are still in the realm of science-fiction. One day, maybe, but for the present *PhonePro* is about as good as it gets.

What one can do with existing technology is tailor the audio materials to take advantage of the strengths of the voice-mail system, while minimizing its weaknesses.

Since the system cannot recognize correct oral responses or respond to incorrect ones, voice input from the calling student must be limited to those exercises or assignments that can be stored and graded later by an instructor or monitor. Since *PhonePro* can be programmed to save voice input in either a separate database record or a separate voice-mail box for each caller, this presents no particular problem. The program can create a record or mail box for each student that calls in to work on a particular exercise or assignment; the instructor can then listen to and grade the student responses later by using either the *PhonePro* software or a proprietary in-house application designed for that purpose.

Touch-tone responses, on the other hand, can be recognized and acknowledged by the telephony software.. For material that does not necessarily call for an oral response, such as aural comprehension exercises, or multiple choice tests, the student can be prompted to enter a numeric response. This type of work can be graded by the software, the student can be given instant feedback, and the score can be stored for retrieval by the instructor.

In a system of this kind, it is, unfortunately, impractical to attempt work requiring written responses of more than a short word or two. The two-key method of inputting alphabetic responses is slow and clumsy at best. For foreign languages, the accented characters pose a particular problem. Other types of software exist for developing written skills in students. For remote access, these include BBS (Bulletin Board System) software, computer conferencing applications and "groupware", all of which require the student to have access to a microcomputer and a modem or a network connection. Each of these technologies comprises an important aspect of the "virtual" language learning center we are trying to implement in the context of our projected physical facility.

Conclusion

A remote-access, audio learning system such as the one presented here is not ideal, certainly. It is incapable of providing simultaneous aural and visual stimuli, except through the expedient of providing students with a corresponding text. The possibilities for oral response are also quite limited, mainly because of the primitive state of voice-recognition software. Consequently, such a system is

ERIC

currently best suited to comprehension activities, such as training in listening and in phonetic discrimination. As a result, the proposed remote-access, audio system is intended to complement, rather than replace, the main language learning center facility. Once public telephone systems make widespread use of digital sound technology and fully implement the ISDN systems now under development, remote learning systems incorporating both audio and video will surely become not only a reality, but a commonplace. Until that happy day dawns, perhaps around the turn of the century in some parts of North America, the system outlined here can provide a viable and cost-effective alternative for at least part of the work now done by students in more traditional, and perhaps less suitable, language lab environments.

References

Blake, R.J. (1987). CALL and the Language Lab of the Future. *ADFL Bulletin*, 18(2), 25-29.

Cammish, H.K. (1983). The Audio-Lingual and Audio-Visual "Revolution." *Teaching Modern Languages*. G. Richardson (ed.), 53-74. London and Canberra: Croom Helm.

Claudel, C.A. (1968). The Language Laboratory. *The Study of Foreign Languages*. J.S. Roucek (ed.), 219-236. N.Y.: The Philosophical Library.

Courchêne, R.J., Glidden, J.I., St. John, J., & Thérien, C. (eds.) (1992). *Comprehension-Based Second Language Teaching/L'Enseignement des langues secondes axé sur la compréhension*. Ottawa: University of Ottawa Press.

Hammerly, H. (1982). *Synthesis in Second Language Teaching: An Introduction to Languistics.* Blaine, Wash. & N. Burnaby, B.C.: Second Language Publications.

Hutchinson, J.C. (1966). The Language Laboratory: Equipment and Utilization. *Trends in Language Teaching*. A. Valdman (ed.). N.Y.: McGraw-Hill.

Iandoli, L.J. (1990). CALL and the Profession: The Current State. *The French Review*, 64(2), 261-272.

Mueller, M., McCavana, G., Ramsden, M., & Shelly, S. (1987). Language Learning Laboratories: The End of a Lukewarm Affair? *The French Review*, 60(5), 585-591.

Nimmons, P.B. (1982). What Can Be Done With a Language Lab? *ADFL Bulletin*, 14(1), 28-32.

Paramskas, D.M. and Thomas, A. (1986). Le Laboratoire de langues à l'heure des approches communicatives. *Mosaïque II: Tendances et pratiques actuelles en didactique du français langue seconde*. Bourbeau-Walker, M., Paramskas, D.M. & Surridge, M. (eds.). L'Association des Professeurs de Français des Universités et Collèges Canadiens.

Politzer, R.L. (1965). *Teaching French: An Introduction to Applied Linguistics*. Second edition. Waltham, Mass.: Blaisdell Publishing Company.

Pusack, J.P. (1986). Computers, Technology, and the Foreign Language Department. *ADFL Bulletin*, 17(3), 46-50.

Rivers, W.M. (1981). *Teaching Foreign-Language Skills, Second Edition*. Chicago and London: The University of Chicago Press.

Stack, E.M. (1966). *The language laboratory and modern language teaching*. Revised edition. N.Y.: Oxford University Press.

Is Browsing Really Instructional? The Use of Hypertext as a Supportive Device in Incidental Learning

LORILEE M. SADLER
Office of Information Technology
Indiana University-Bloomington, IN 47405, USA

Incidental learning is the kind of learning that happens in every day life. It is that unexpected experience that occurs when, while concentrating on one thing, you learn another. Incidental learning can be defined as any learning that takes place during the active process of learning something else. It is curious to examine the informational tools that each of us has and to question their origins. Many people will claim that the majority of their knowledge was learned not in the classroom but rather through the trials and errors of day-to-day life, through overheard conversations, and through the searches for information that lead one down paths that, while not necessarily relevant to the task, were interesting nonetheless (Postman, 1964). Historically, incidental learning has been disdainfully viewed as unimportant by educators (Postman, 1964). This is most likely because of the threat that this kind of learning poses to instructors. However, within the last fifteen years studies of incidental learning and the acquisition and retention of skills have increased. The notion of incidental learning has risen in the paradigm hierarchy from something that was invalid as a method of learning to an entity which is both legitimate and ubiquitous

The role of traditional education is to provide on-task content so that specific learning objectives can be met in a pre-determined time period. Incidental learning in traditional education is, at best, considered a serendipitous phenomenon and, at worst, left completely to the non-school world to provide. Educational philosophers argue that what is being taught in school is not necessarily what needs to be known to survive in the real world (Bransford, in press) and that new technologies such as hypertext can support epistemologies of learning that provide students not with "factual" educations but rather with abilities to connect facts and ideas to solve problems and to think analytically (Brown, 1988).

The advent of hypertext as a popular computing tool has many implications on the role of incidental learning in formal education. Educational research and popular journals alike are printing articles dealing with hypertext and peripheral subjects in almost every issue. However, many articles deal with specific projects using hypertext packages and focus on the "gee whiz" aspects of the technology of hypertext rather than the actual instructional value of the application. Others describe how easy it is to develop computer-assisted-instruction-like materials using hypertext. In both cases the real value of hypertext systems has been overlooked. Little work is being done to determine the types of environments, learning, or students that hypertext supports.

The notion of hypertext is one that is not new. In 1945 Vannevar Bush conceptualized a machine that would enable users to view several documents at one time and to connect those documents in a manner that was meaningful to each individual user. Ted Nelson (1974) began work on a system called Xanadu in 1965 that he hoped would realize Bush's dream machine. It was not until the late 1980's, however, that computer technology became affordable enough to support hypertext systems on a variety of platforms. Today, hypertext systems exist for most types of computers.

Although Bush provided only a schema for hypertext, most experts agree on a few critical attributes. A hypertext system is one that stores information in small chunks called nodes. Since these nodes are organized in a non-linear fashion, system developers or users may create links between nodes to relate information in some specific way. A set of linked nodes forms a path which represents an individual's network of ideas. A hypertext system is rarely linear even when many nodes are linked together. Systems which contain developer-defined links are said to have pre-defined links through which users can navigate. In these systems the user almost always has the ability to choose among several paths to follow and can follow one or all of them. Moreover, users can also jump from one path to another at any given time.

In direct contrast to computer-based instruction systems, true hypertext systems contain no specifically imbedded learning goals. Instead, they strive to facilitate multi-causal thinking. Information is provided in an information base and it is the responsibility of the user or supervisor of the user to impose a learning goal. While the information base of a hypertext system is generally the heart of the system, hypertext systems can contain other modules that can be made to look like a CBI lesson, an expert system, or an intelligent tutor. These modules are generally peripheral to the information base rather than inherent to it. Many hypertext applications enable the user to define paths by making links between information nodes. This feature of hypertext may well be the most important feature of the entire system since it challenges the user to make associative links between information to form a customized idea network.

These attributes of hypertext systems differ greatly from traditional computer-based instruction (CBI) in which on-taskness is one of the primary goals of the system. In CBI systems a learning objective is built into the system. This forces the user to see and do every part of the lesson that the designer deems necessary. Often the result is that the user becomes bored with or ambivalent to the material being presented. Traditional CBI includes content questions at the end of each unit. Students who have a moderate amount of experience with CBI instinctively pick out content that will be measured at the end of the module. Many of the same problems exist in traditional print text. Because they often focus on the primary information, learners frequently miss the subtleties or implicit information within the text.

Hypertext systems support incidental learning in several ways. First, users have direct access to large information bases that can be traversed quickly and in a variety of ways. Second, the process of active browsing through a hypertext information base allows users to see many different types of information. Users will often encounter information that is not relevant to the task at hand but that does have some relational or personal relevance. Finally, hypertext systems enable users to directly experience and manipulate multiple representations of information.

Through this process users not only acquire information but also experience the relationships that exist between information nodes. This direct manipulation of relationships causes users to be able to generalize information across situations (Spiro, 1988).

Studies of the relationship between hypertext and incidental learning are almost non-existent. Hypertext, because it is a new technology, has seen a great deal of press over the last three years, but researchers are only now dealing with pedagogical issues surrounding this technology. New pedagogical philosophies can be readily supported by hypertext environments, but much research needs to be done to gain evidence of the degree of effectiveness of both hypertext as a facilitator of these new theories as well as the theories themselves. In this experiment, the relationship between hypertext and incidental learning is explored. When asked to create a self-imposed task and then follow through with that task, learners acquire information related to that task as well as significant chunks of incidental information while browsing. Learners who are given an experimenter-defined task should be able to acquire as much of the information as the task itself asks for, but their acquisition of incidental information will be much lower because of their concentration on the task mandated by the experimenter.

Hypertext systems can provide enriched learning environments that differ greatly from any other learning environment that has previously been used in education. Hypertext systems provide learners with multiple dimensions of information. Each dimension or path eschews a different perspective on not only the information being presented but also the relationships developed through multiple-path navigation. Users of hypertext systems see not just one representation of information but multiple representations. It is these multiple representations that aid users in utilizing what they have learned in many different situations (Spiro, 1988). One reason that multiple-representations can be developed in hypertext systems is that the speed at which idea networks are developed on the computer can rival those of the human mind. This spontaneous thought that occurs in a hypertext system enables each user to freely develop interest paths without being distracted by waiting periods that occur on the system. In creating or following multiple representations of information, users determine that which is relevant and irrelevant information in an information base.

This experiment has been designed to test the degree of incidental learning in a hypertext environment when users are given different tasks. In the experiment subjects are assigned one of two tasks: one group will be asked to pursue avenues that interest them personally by freely browsing through the system, and to identify any self-imposed tasks at the end of the browse period; the second group will be given a scavenger hunt scenario and a list of facts and objects to find throughout the system. All subjects will browse the same hypertext application to complete their tasks and will be limited to twenty-five minutes within the system. Subjects are told that questions over the text will be asked when they have finished their task.

The post-browsing questions are given in two parts. The first part of the posttest evaluates the acquisition of relational information by asking subjects to complete a partially drawn map of Glasgow, Scotland. The second part of the posttest is a set of multiple-choice questions designed to measure the acquisition of "factual" information such as dates and historic events relative to Glasgow.

Methods

Subjects

Twenty-four subjects were involved in this experiment: four in a pilot test of the experiment and twenty in the actual experiment test. Subjects were divided equally, through a drawing system, into each of the two experimental groups. Subjects were graduate students from the computer science department at Indiana University and participated solely on a volunteer basis. Subjects ranged in age from twenty-four to thirty years of age. Thirteen of the subjects were natural-born Americans, 1 subject was Korean, 6 subjects were Indian, and the remaining four subjects were from mainland China. Subjects were run during a four-day period at times that met with their convenience. All subjects had used Macintosh computers prior to the experiment and were familiar with the operating system and eight of the subjects had taught an introductory computer tools course using the same hypertext development system that was used to develop the hypertext application for this experiment..

Materials

A hypertext system called Glasgow On-Line was commissioned by the city of Glasgow, Scotland for use by tourists. This system was developed in HyperCard, a hypertext generation product for the Macintosh computer. Glasgow On-Line contains maps of the city, indexes of

ERIC

restaurants, museums, shopping places, physicians, banks, and other items of interest, as well as factual information about important city landmarks. The system also contains a powerful "find" utility that enables users to search for a specific shop or product. The Glasgow On-Line system was run, for the experiment, on a Macintosh IIx with a color monitor, extended Apple keyboard, and a single-button Apple mouse.

Procedure

Prior to running the actual experiment a pilot test was done using four subjects. Four pieces of paper, two displaying the letter "A" and two displaying the letter "B" were placed into a small box. Upon entering the experiment room, each subject drew a piece of paper from the box. Subjects belonging to group A were asked to freely browse the hypertext system in its entirety, imposing their own task if they wished. Subjects belonging to group B were presented with a list of items to locate within the city of Glasgow, as in a scavenger hunt. Neither group was allowed to make notations of any kind. After subjects completed their twenty-five minutes of browsing or searching they were given the two-part posttest described above. It was determined during the pilot test that the scope of the entire information base was much too large for the amount of time and information requested on the posttest. Subjects in group A had great difficulty deciding where to start in their browsing and changed self-imposed tasks often. They were unable to answer most of the factual questions on the posttest and were able to demonstrate little relational information on the posttest. Subjects in group B were able to navigate the information base to find items on the scavenger hunt list, but spent little or no time exploring alternate avenues or information that was peripheral to their task. These initial findings have implications for future research and will be examined in the discussion portion of this paper. The immediate effect of these findings was that the scope of the information base used during the experiment was modified to include only one quadrant of the city of Glasgow and the information that fell within that quadrant.

For the actual experiment, subjects were divided into groups based on the algorithm used in the pilot test. Twenty-four pieces of paper, divided into two sets of twelve with each set labeled "A" or "B, were placed in a box and mixed together. Upon entering the experiment room on the experiment day each subject drew a lettered piece of paper from the box. Subjects drawing letter "A" were asked to browse through the on-line system for twenty-five minutes, looking at things that were interesting to them. Subjects in group A were told that if they created a task for themselves (e.g. they looked for every veterinarian in the assigned quadrant of Glasgow) that they should write this task down on a piece of paper provided by the experimenter. Subjects drawing letter "B" were given a scavenger hunt scenario to work through using the Glasgow On-Line hypertext application. All subjects were instructed to spend twenty-five minutes browsing through the application and to answer some simple questions after that time. Subjects were required to sit at the computer workstation for the entire twenty-five minutes. This requirement was imposed so that the time variable was controlled for and also to encourage random browsing in both groups. After twenty-five minutes of browsing, the each subject took the posttest. Subjects were seated away from the computer for the test and were not allowed to look back at the computer screen or to refer to any part of the on-line system. As stated earlier, subjects were not allowed to take notes.

Results

Results for the study are shown in Table 1. Scores for the relational and functional portions of the posttest were kept separate for each subject in each group. The mean for the relational portion of the posttest for the no task group was 9.9 and for the scavenger hunt group was 6.1.

Comparison of these means shows that the no task group had a mean that was 3.8 points greater than the mean of the scavenger hunt group. The mean for the factual portion of the posttest for the no task group was 6.5 and for the scavenger hunt group was 8.5. A 2.0 point higher achievement for the scavenger hunt group is indicated by these results. Although further statistical analysis is necessary to determine significance, the initial means do indicate some difference between groups. These findings are in concurrence with the hypothesis concerning non-task versus task-driven uses in browsing a hypertext system.

Table 1
Results of Posttest

	Subjects 1	2	3	4	5	6	7	8	9	10	Mean
Group A											
Relational	8	6	12	15	10	13	10	9	8	8	9.9
Factual	7	8	6	4	7	9	10	6	3	5	6.5
GroupB											
Relational	7	7	10	5	10	3	4	5	5	5	6.1
Factual	9	6	8	7	10	8	9	9	10	9	8.5

Discussion

Results yielded from both the pilot study and the actual experiment have proven to be informational and give rise to new questions. The original hypothesis was that learners given no specific task other than to browse a hypertext system for their own interests would gain more relational information than would those learners who were directed in a specific "search" task, and that learners who were given no specific task would concentrate less on factual information given in the system than would specific task learners. In the pilot study it was obvious to the experimenter that the scope of the information base being used was much too large for the time constraints placed on the subjects and much too broad for subjects who had no experimenter-defined task. This observation needs to be examined under carefully controlled conditions in further research to see if size and breadth of a hypertext system are factors in learning through browsing.

In the actual experiment, subjects had more than enough time to browse freely and find the information they were supposed to find. Because the portion of the hypertext system used was not heavily saturated with hyper areas (hot spots), subjects who were not assigned a specific task (those in group A) were able to explore each of the hyper areas on the screen and thus did not impose their own search task. This was somewhat of a surprise to the experimenter and this finding too is cause for another investigation concerning saturation of hyper areas and motivation to explore.

Hypertext systems have opened new doors to educators and learners. It is important that everyone involved in the design, development, and delivery of instruction examine closely the capabilities of this new technology and force the technology to support educational aims rather than allow education to support the new technology. More research needs to be done in the area of incidental learning and hypertext to determine the extent which hypertext supports this kind of learning and to determine, in more concrete terms, the importance of incidental learning.

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. In D. Nix & R. Spiro (Eds.), Cognition, education, and multimedia: exploring ideas in high technology Hillsdale, NJ: Erlbaum.

Derry, S. J. (1984, April). Strategy training: An incidental learning model for CAI. Paper presented at the *Annual Meeting of the American Educational Research Association*, New Orleans, LA.

Duffy, T.M. & Knuth, R.A. (1989, July). Hypermedia and Instruction: Where is the match? Paper presented at the *NATO conference Designing Hypertext for Learning*, Tubingen, West Germany.

Gagne, R. M. (1986). Instructional Technology: The research field. *Journal of Instructional Development*, 8, 7-14.

Gamberg, R. & Others. (1988). *Learning and Loving It. Theme Studies in the Classroom.* Portsmouth, NH: Heinemann Educational Books, Inc.

Nelson, T.H. (1974). Dream Machines. Chicago: Author.

Olds, H. F. (1982, July). The microcomputer--an environment that teaches: Exploring the hidden curriculum. Paper presented at the *Annual Summer Conference*, Eugene, Oregon.

Postman, L. (1964). *Short-term memory and incidental learning.* New York: Academic Press.

Spiro, R. J., Coulson, R. L., Feltovich, P. J., & Anderson, D. K. (1988). Cognitive flexibility theory: Advanced knowledge acquisition in ill-structured domains (Tech. Rep. No. 441). University of Illinois at Urbana-Champaigne, Center for the Study of Reading.

ERIC

MOTIVATING UNIVERSITY FACULTY TO INTEGRATE MULTIMEDIA INTO CLASSROOM PRESENTATIONS

MARTHA C. SAMMONS
Professor, Department of English
Wright State University, Dayton OH 45435

Purpose

This paper describes a plan to motivate faculty to use multimedia for classroom presentations to improve teaching and learning throughout the undergraduate curriculum at Wright State University. The purpose of the project is to incorporate multimedia as an integral part of classroom lectures through an **aggressive, strategic, university-wide** plan. (Multimedia presentations are defined here as the integration of computer-generated course outlines and notes with graphics, video and sound). This paper will discuss the philosophy and strategy of the plan to motivate faculty, including ways we implemented the plan despite severe state budget cuts.

Background

Past initiatives: Exploration of the feasibility of launching a major multimedia initiative at Wright State began just three years ago. Twenty-two faculty currently use multimedia presentations in their classes. As a result, several schools have visited our campus to learn about our multimedia work.

A Multimedia White Paper published by the Department of Academic Affairs developed into a detailed strategic plan for implementing multimedia in the classroom. This plan identified both long-term and short-term tasks. I was given a leave to assume the role of Faculty Liaison to the multimedia project. Among my responsibilities are forging linkages between this project and other computer/technology committees on campus, seeking funding, and developing training materials and the faculty curriculum. I am also conducting a pilot program in the College of Liberal Arts.

Future initiatives: Future initiatives involve conducting this pilot program and establishing multimedia as part of a teaching center.

A pilot program in the College of Liberal Arts will give notebook computers to faculty who proposed innovative ways to use computer technology in the classroom with the aid of projection and other peripheral equipment. This project will evaluate the effectiveness of a faculty task force in directing the project. This committee will assess the effectiveness of both new classroom methodology and various new technologies.

Second, multimedia will be transferred from Computing Services to our new Center for Teaching and Learning, which will foster a "teaching culture" that continuously improves teaching and assesses ways to improve student learning. The Center emphasizes the total support of the administration to teaching innovation and excellence. The primary goal of this multimedia support group is to consult and coordinate: to encourage and train faculty to use multimedia, provide technical expertise, and serve as a consulting group for each college. In addition, the Center will coordinate development of presentations and training materials and provide access to equipment and resources available both on and off campus. With funding, the Center will eventually support innovative teaching through incentives such as merit pay, leaves and other opportunities to develop multimedia materials.

Problems in Motivating Faculty

Higher education has lagged behind primary and secondary education in incorporating multimedia into both teaching and learning. Yet numerous studies, including our own, show increased effectiveness of presenting classroom material through a variety of media. While a few major universities--IUPUI, University of Delaware, Washington State, etc.--are beginning to explore uses of multimedia, most have concentrated on development of courseware, computer-based instruction, and computer laboratories for student instruction.

Our project focuses on faculty use of multimedia as a way to enhance classroom lecturing and thus teaching and student learning. The literature has established that college faculty in general are slow to integrate new technology into the instructional process. Studies also indicate that the traditional faculty lecture involves mainly use of the blackboard, overheads, and occasional slides.

We believe faculty are reluctant to use multimedia for the following reasons:

1. **Knowledge and Training**
   - unawareness about how multimedia can enhance teaching and learning in the classroom
   - unawareness about ways to tailor multimedia to a specific discipline or integrate multimedia into the current curriculum
   - inability to use multimedia hardware and software
   - perception that there will be difficulties or disadvantages that arise in actual classroom implementation of new technology
   - uncertainty about how to gather appropriate materials for use in multimedia presentations
   - lack of training materials specifically geared to teachers
2. **Lack of Control**
   - skepticism about technology when technicians rather than faculty direct its use
   - failure of projects in which technology rather than the material dictates the medium
3. **Time**
   - commitment of time needed for multimedia training
   - long development time for creation of multimedia presentations
4. **Incentives**
   - lack of administrative support for multimedia
   - failure of universities to incorporate innovative teaching as part of the merit system
5. **Money**
   - lack of money for new hardware and software

Strategies for Motivating Faculty

Based on our experience thus far, we offer the following ten strategies for motivating faculty to use multimedia:

1. **Show leadership and organization**

Faculty must see administrative support for their efforts. Administrative commitment is also vital to appropriate necessary funds for equipment and an incentives program. In addition, there must be a strategic plan for incorporating multimedia in the classroom. Otherwise, there will be only piecemeal efforts by a handful of pioneer faculty. There must also be support from the computer support personnel and faculty leaders.

2. Educate

Faculty must first see the possibilities available to them, as well as the benefits of using technology in the classroom. This is accomplished by technology fairs, conferences, and special speakers. They must also see how easily and efficiently they can build course materials using the computer.

Faculty also often fail to see the value of multimedia because they do not have a clear idea of how it can be effectively used in their specific disciplines. Seminars within each college allow guest faculty both from within the college and from other universities to demonstrate ways multimedia can be used in specific disciplines. Faculty must then continue to be informed of developments in their field.

In addition, the goal of the multimedia project is to improve learning through enhanced teaching methods and through increased student access to a variety of information. Faculty will be motivated when they are offered the possibility of using more effective instructional approaches and see the results of enhanced student learning. They will be challenged by assuming a new role that is better for them and their students.

3. Train

Faculty must be trained to use technology. Thus a formal training program for faculty allows faculty members to go through distinct **phases** in their progression to use of full multimedia in the classroom. We recognize that not all faculty nor all courses benefit from progressions through all phases, but that most benefit from at least the early steps. These courses are taught by faculty or staff with expertise in a specific subject.

Phase 1 involves training in basic techniques, such as simple slide presentations, and videodisc/CD players for incorporating pictures and sound segments in class. This training makes faculty comfortable with computers, presentation programs, easy-to-use presentation equipment, and simple processes for digitizing images and/or sound. Phase 2 involves training in more sophisticated uses of multimedia in formal multimedia classrooms, advanced authoring, and courseware design.

Workshops: Workshops are designed to fit into the typical faculty schedules. A series of 1-2 hour hands-on, interactive, on-going workshops allows a faculty member to specialize in technology appropriate to his/her field and produce something that can be used in class. Workshop topics fall into the three main areas: development, software, and hardware.

Training materials: Training in multimedia technology will not be successful without adequate materials geared toward teachers. We are thus developing training materials geared specifically to faculty. Assessment is also a major emphasis of the training program. Much of the assessment data that has been presented thus far in the literature has focused on the effectiveness of multimedia in K-12 teaching, teacher education programs, and the effectiveness of interactive videodisc as a training tool.

We will gather data about the effectiveness of these areas:

- **methods of training faculty** in use of multimedia. We will test the effectiveness of a variety of teaching techniques and materials, as well as the effectiveness of training in motivating faculty.
- **the effect of multimedia on student learning**. We have begun to gather data from

current multimedia classes about enhancing learning through multi-sensory and interactive presentations. We are developing appropriate assessment materials, as well as guidelines for administering these materials.

4. Let faculty play a central role in decision-making

Faculty must participate in the process of deciding how technology will be used before they will adopt it. The College of Liberal Arts pilot project will test the effectiveness of a Task Force in administering a technology plan. This committee, comprised of members of each constituency, will decide on the incentives program and assess its effectiveness.

5. Offer a variety of technology options

Realizing that all faculty are not ready for multimedia, we are investing in prototypes of new technologies. This technology includes the following:

- notebook computers
- LCD projectors
- scan converters for projection of notebook computer displays on TV screens
- videodisc players with barcode readers and barcode software
- CD-ROMs and readers
- in-house or external mastering of CD-ROM's and videodiscs

Faculty also need to see that this varied technology will support individualized teaching methods, not dictate it. New technologies are demonstrated at technology fairs; then faculty are encouraged to try using them in class. Faculty successes with this equipment will dictate future investments in technology.

6. Begin with the simple to allow growth and flexibility

While some faculty were quick to adopt multimedia, the majority were not. Thus a faculty development program must accommodate different levels of experience with technology, as well as fundamental differences in the disciplines. Changing instructional methods must be evolutionary or incremental. With the options listed above, faculty can begin with either simple slide presentations or videodisc players. They can then grow into other options as they are ready.

7. Offer optional classroom environments

We are creating two types of classrooms: large multimedia classrooms with permanent equipment and portable configurations for smaller classrooms. We have invested over $100,000 in multimedia equipment and currently own four workstations for creating presentations and four "presentation" stations used in the classroon We are renovating five large classrooms for multimedia.

We are also making available an assortment of portable multimedia carts with various equipment configurations in order to "down-size" multimedia for smaller groups. The faculty member, for example, can simply bring a notebook computer to class and use the TV monitors already in place.

Regardless of the type of classroom environment, faculty want equipment that is simple to learn, easy to operate, instantly available, reliable, and powerful.

8. Offer continued support

Support must not only come from administration but also from technical personnel and peers. Faculty should be trained to use and trouble-shoot equipment, but technical help should be immediately available. In addition, user groups are another source of encouragement. Faculty teaching common subject matter to should be encouraged to collaborate to develop a standard database of resource materials. Faculty should also be encouraged to share their visions and successes with others in their departments. Furthermore, faculty can find inspiration and a new source of common interest with members of other departments.

ERIC

9. Help speed development

Despite good equipment and training, many faculty will not use multimedia if it continues to require 5-10 hours of development time per lecture. We thus encourage the tailoring of multimedia presentations to a wide variety of teaching styles and use a team approach to planning and designing presentations. This design team assists faculty in determining the appropriate media for their discipline (text charts, graphs, tables, diagrams, photos, graphics, animation, videotape, videodisc, sound). Then in an effort to move away from traditional college lecture techniques, the team demonstrates and suggests ways to use technology to support the chosen media (computerized screen shows, videodisc/CD ROM segments, full multimedia, etc.).

We are also exploring ways to speed authoring of course materials through

● establishing a **materials lab** for digitizing sounds and images. This lab is staffed by student assistants. We are also seeking ways to use graduate students within each discipline to assist in gathering and preparing appropriate presentation materials.

● developing clear written **policies and procedures** for submitting materials to the staff. Faculty should know what is required of them, as well as the turn-around time they can expect.

● developing a **database** of these materials for distribution. We investigating the potential of in-house CD-ROM publishing for storage of materials for the large general education classes.

● establishing **equipment and software standards** within the university to simplify training and support.

Development is also aided by giving faculty access to the tools and resources they need to create their own presentations. "Creation" workstations should be located in a convenient and friendly environment.

10. Offer an incentives program

Faculty are unmotivated to use multimedia because of the lack of reward for innovative teaching and development of course materials and courseware in the traditional merit system. Rewards can take the form of money or equipment, stipends, mini-grants, development leaves, travel funding, merit pay, or public recognition in publications and conferences.

In the College of Liberal Arts pilot project, we will offer notebook computers to faculty who submit winning proposals about how they will bring technology into the classroom. This rewards faculty for rethinking their teaching methods and course materials.

Summary

The multimedia project at Wright State is

● an aggressive, university-wide approach to incorporating multimedia in the classroom

● focuses on improving teaching and learning through lectures enhanced with computerized audio-visual materials

In establishing this project, we have learned that to motivate faculty to use the equipment and change their establish teaching styles, faculty must have strong administrative support, be educated about the value and potential of multimedia, receive on-going training, play a central role in decision-making, have a variety of technology options, be allowed to grow technologically, have a choice of teaching environments, be given continued technical and peer support, have assistance and adequate resources for development, and be offered incentives to change their current teaching methods.

References

Ed. Anandam, Kamala (1989). *Transforming Teaching with Technology: Perspectives from Two-Year Colleges.* EDUCOM/Academic Computing.

Bruder, Isabelle (1991). "Guide to Multimedia: How It Changes the Way We Learn and Teach." *Electronic Learning*, Sept., 22-26.

Elmore, Garland (1991). "Planning and Developing a Multimedia Learning Environment." *THE Journal*, 18 (7), 83-8.

FIPSE Technology-Study Group (1989). *Ivory Towers, Silicon Basements: Learner-Centered Computing in Postsecondary Education.* EDUCOM/Academic Computing.

Heermann, Barry (1988). *Teaching & Learning With Computers*. Jossey-Bass.

Monaghan, Peter (1993). "Multimedia Teaching Techniques Gain Momentum at Washington State University." *Chronicle of Higher Education*, Feb. 24, A19-21.

Reisman, S. & Carr, W.A. (1991). "Perspectives on Multimedia Systems in Education." *IBM Systems Journal*, 30 (3) 280-96.

Teaching With Computers (1991). Apple Computer.

HyperGlyphs: Using Design and Language to Define Hypermedia Navigation

PATRICIA SEARCH
Department of Language, Literature, and Communication
Rensselaer Polytechnic Institute
Troy, NY 12180, USA

Hypermedia is an audiovisual medium that presents new challenges in information design. Today, most hypermedia applications are created for specific tasks within a given culture. However, in the global networks of the future, hypermedia interfaces will establish bridges between cultures by using audiovisual symbols to define diverse cultural perspectives.

In order to expand the communication potential of hypermedia computing, we need to develop interfaces with orientation cues that help users navigate through large, multimedia databases. In a hardcopy document, spatial references, such as the location of information on the page or the sequential position of information within the document, help the user locate specific text and images. However, hypermedia computing is a temporal medium in which spatial relationships change dynamically, leaving the user with few references for orientation.

At Rensselaer Polytechnic Institute, I am directing a project called HyperGlyphs which is evaluating criteria for hypermedia interface design. The prototype program for this project is a hypermedia program on contemporary art. A focus group of artists, art historians, and art curators, trained to analyze the impact of design on the communication process, is specifying the requirements for the user interface. Although these professionals have little computer experience, their training in audiovisual communication enables them to analyze the psychodynamics of the medium and identify the types of perceptual cues that should be used for orientation references.

The database for this project, which consists of abstract images and text, is an important element in the research design. By using abstract imagery rather than representational data, it is possible to focus on the perceptual and cognitive relationships between pictorial symbols and words. As a result, it is easier to identify new levels of interaction between perception and cognition that can define orientation cues for hypermedia navigation.

The project began with a focus-group study that included 60 hours of interviews that defined the user's tasks, the objectives of the program, and the design of the user interface. This study produced approximately 300 iterations of screen designs that were developed on paper before any software was written. The research team then used this conceptual design to develop a prototype program.

The HyperGlyphs project is focusing on how graphic design and language in the user interface can facilitate navigation and information retrieval. Since all hypermedia programs include interactive functions that allow the user to access information in a multimedia database, the results of this study can be applied to a wide range of hypermedia applications for schools, industry, government, and specialized professions.

Criteria for Hypermedia Interfaces

Hypermedia interfaces can reduce the potential for confusion and disorientation if they encourage the user to structure a cognitive approach to information retrieval. This approach does not preclude browsing through a multimedia database. Instead, it means the hypermedia interface should help the user evaluate the contents of the program and determine the best ways to use the

ERIC

program to accomplish specific tasks. To enable the user to meet these two objectives, a hypermedia interface should do the following:

- Acknowledge the limitations of the medium;
- Identify the levels of authorship in the program;
- Define and visualize the semiotic model for the program;
- Provide spatiotemporal overviews of the database and individual references;
- Narrow the objectives of navigation functions that track and visualize the user's path.

This paper discusses each of these points and uses examples from the HyperGlyphs Program to illustrate how graphic design and language can help the user become an active rather than passive participant in the navigation process.

Acknowledge the Limitations of the Medium

The introduction of hypermedia computing has been accompanied by unrealistic expectations and numerous misconceptions about the power and capabilities of the medium. To begin, there is a tendency to view hypermedia programs as all-inclusive, authoritative databases of information that eliminate the need for external resources. However, most hypermedia projects do not include large, comprehensive databases that fulfill all the research and professional needs of the user. Copyright restrictions, the lack of industry standards for exchanging data, and the high cost of hardware and software powerful enough to store and manage large quantities of audiovisual information have made it extremely difficult for the developers of multimedia programs to assemble this type of idealistic database.

To dispel this myth about hypermedia, the user interface should clearly delineate the physical and conceptual limitations of the database and encourage the user to evaluate additional resources to augment the contents of the program. The interface should also remind the user about the perceptual limitations of the medium. Hypermedia programs provide important audiovisual feedback for the user, but they do not provide spatial and tactile references--sensory information that may be required for certain tasks. If users are reminded of these limitations, they can consciously evaluate their information needs and develop structured, cognitive approaches to navigation that reduce confusion.

A brief introduction or tour of a hypermedia program is a good place to include a statement that cautions users about the limitations of the medium. An introductory statement or graphic should summarize the contents of the database and recommend additional references on a subject. Language, in turn, can define the perceptual limitations of the database. Terms like *digital reproduction*, *digitized text*, and *video excerpt* remind the user that electronic technology is a perceptual filter that limits sensory feedback, and therefore, it is important to consciously evaluate the merits of each reference for specific informational needs.

Identify the Multiple Levels of Authorship

Hypermedia programs contain several levels of authorship including the designers or editors of the program, the authors of the individual references in the database, and the users themselves who create links between the references (Search, 1992). Many hypermedia programs do not define these levels of authorship. For example, it is often difficult to locate the names and credentials of the designers or editors of the program. In some programs, users can move from reference to reference without ever knowing the authors or the sources of the material. This problem often occurs when users follow trails of "hotwords" from one reference to another.

An interface that clearly delineates the different levels of authorship in a hypermedia program helps the user understand the structure and objectives of the program and provides orientation flags for navigation and information retrieval. Once again users can easily evaluate their information needs and define cognitive approaches to navigation that reduce confusion and disorientation.

There are several ways to define these levels of authorship and improve navigation. To begin, the introduction or tour of the program should highlight the names and credentials of the designers and editors of the program. The introduction should also include a statement that explains the objectives and structure of the program.

In addition, the program should clearly identify the author, title, publisher, and date of each reference in the database. Similar citations should accompany still images or motion video. In the HyperGlyphs project, for example, when the user displays information on the screen, a bibliographic citation appears in a header at the top of the screen. This citation provides the user with feedback about the editorial perspectives and the organization of the database. The bibliographic references also establish orientation labels that help the user structure and locate information (Wright, 1991).

Finally, hypermedia programs should include interactive tools that help the user restructure information and author new relationships. Tools for creating bookmarks, annotations, timelines, organizational charts, and new links in the database increase the user's awareness of the conceptual organization of the database and help the user restructure the information to meet personal objectives. Once again, this type of interaction enables the user to develop a cognitive approach to navigation that reduces the problems of "decentering" and fragmentation that often cause confusion and disorientation in hypermedia programs (Landow, 1989, p. 185; Mullins, 1991).

Define and Visualize the Semiotic Model for the Program

A well-defined semiotic model for the hypermedia program is also an important criterion for successful navigation. The semantic structure of the program and the syntax of the interactive functions should be accurately mapped to a conceptual model of the user's tasks. Since users must be able to browse as well as search for specific information, this semiotic model must be flexible enough to encourage free association between ideas and at the same time, provide a clearly articulated, cognitive framework that facilitates information retrieval.

To create this type of user interface, it is important to identify the cognitive relationships between individual tasks and then map those relationships to functions that organize the database into small, task-oriented groups of data. This type of relational model establishes labels for information retrieval and creates a semantic structure that improves the effectiveness and efficiency of user navigation in a large database of information (Hara, Keller, & Wiederhold, 1991).

In the HyperGlyphs project, the focus group identified the types of tasks to be performed, mapped each task to a set of procedures, and used cognitive walkthroughs (Lewis, Polson, Wharton, & Rieman, 1990) to evaluate and modify the semantic structure of the database and the syntax of the program functions. Throughout this iterative process, the focus group used design and language to visualize this semantic-syntactic model. These visualization techniques helped the focus group clarify the hierarchial and associative relationships in the program and identify the most effective ways to use design and language as navigational aids.

The HyperGlyphs program consists of four top-level categories: Artwork, Exhibits, Bibliography, and Biography. Within each of these sections, the user can navigate through the database by sorting the information according to predefined categories or by conducting keyword searches for specific data. This approach reduces confusion because it divides the database into task-oriented groups of data that establish landmarks for casual browsing and labels for keyword searches.

In the screen designs, color coding is an important navigational aid that creates a visual outline of the database structure. The background color for the interface screens is gray; the four top-level or main categories in the program are represented by color-coded icons. When a category is selected, the icon for that section is highlighted in color while the other three icons remain gray. This color coding is repeated in the second level of icons that appears within each category. The colors of these secondary icons are slightly lighter in value than the main icons to differentiate the two levels of organization. For example, the color code for the "Bibliography" icon is red, and the

"Bookmark" and "Timeline" icons within the Bibliography section are a slightly lighter shade of that red.

The layout of the icons and buttons on the screen also helps the user visualize the semantic-syntactic model for the program. For example, the main icons and the buttons for the sort and search options are arranged in vertical lists that visualize the top-down structure of the program. The lists for the sort and search options are also arranged in parallel columns to symbolize the parallel syntax of the two functions. In addition, these lists are permanent menus (as opposed to pull-down or pop-up menus) that identify the user's location in the database and remind the user how to access information. These permanent menus reduce the cognitive overhead for the user because once information is located in the database, it is not necessary to remember all the steps that are needed to repeat the process.

Research has shown that repetition is an important navigational aid in speech-controlled hypermedia programs (Arons, 1991). Effective audio communication must be redundant because words vanish and information must be repeated in order to maintain the line of thought (Zuboff, 1988). However, the use of repetition as a navigational aid should not be restricted to interactive audio programs. This design technique also creates a permanent frame of reference for navigation in audiovisual interfaces where a dynamic array of perceptual stimuli increases the complexity of the interface (Search, 1993a).

Language is also an important design element that aids navigation by defining paths that help users make choices. Researchers have pointed out that users are less likely to feel disoriented or lost when there is a predefined path that narrows their options for navigation (Zellweger, 1989). As previously mentioned, the HyperGlyphs program provides the user with predefined labels for database sorts and searches. The focus group carefully evaluated and tested the language for these labels because research has shown that successful navigation is linked to the user's ability to interpret the correlation between menu options and information in the database (Barnard, Morton, Long, & Ottley, 1977; Snowberry, Parkinson, & Sisson, 1985).

Similarly, language can reduce confusion and improve navigation if the interface uses linguistic syntax to define the syntax of the program functions. For example, in the HyperGlyphs project, the focus group evaluated the phrases "Search by..." and "Search for..." to serve as headers for the lists of options for the database sorts and searches. The group eventually chose "Search by..." because the phrase tends to limit the user input to a specific list of requests while "Search for..." implies that the user is free to search for a much wider range of information--a process that can lead to confusion and disorientation in the program (Search, 1993a).

Provide Spatiotemporal Overviews

The hypermedia interface should provide overviews that indicate the content and scope of the database. The user can then evaluate the suitability of the database for a particular task and estimate the amount of time and effort needed to navigate through the information. These overviews should specify the organizational hierarchy of the information, the number of resources in the database, and the type of resources (e.g., full-text vs. bibliographic citations, color photographs vs. black and white illustrations, complete video productions vs. video excerpts).

In the HyperGlyphs project, two design techniques provide overviews of the information in the database: graphs and screen headers. Graphs at the beginning of each of the four main categories define the contents of the database by showing how many references are available and whether the textual references are full-text entries or bibliographic citations. Headers at the top of each screen specify the number of entries in the database, the period of time covered by the entries, and the date of the most recent update (see Figure 1). This information gives the reader an idea of the amount of time needed to navigate through the database. Also, by specifying the period of time covered by the entries and the date of the last update, the user is reminded that the database may not be a comprehensive source of information and that other references may be needed to complete a task.

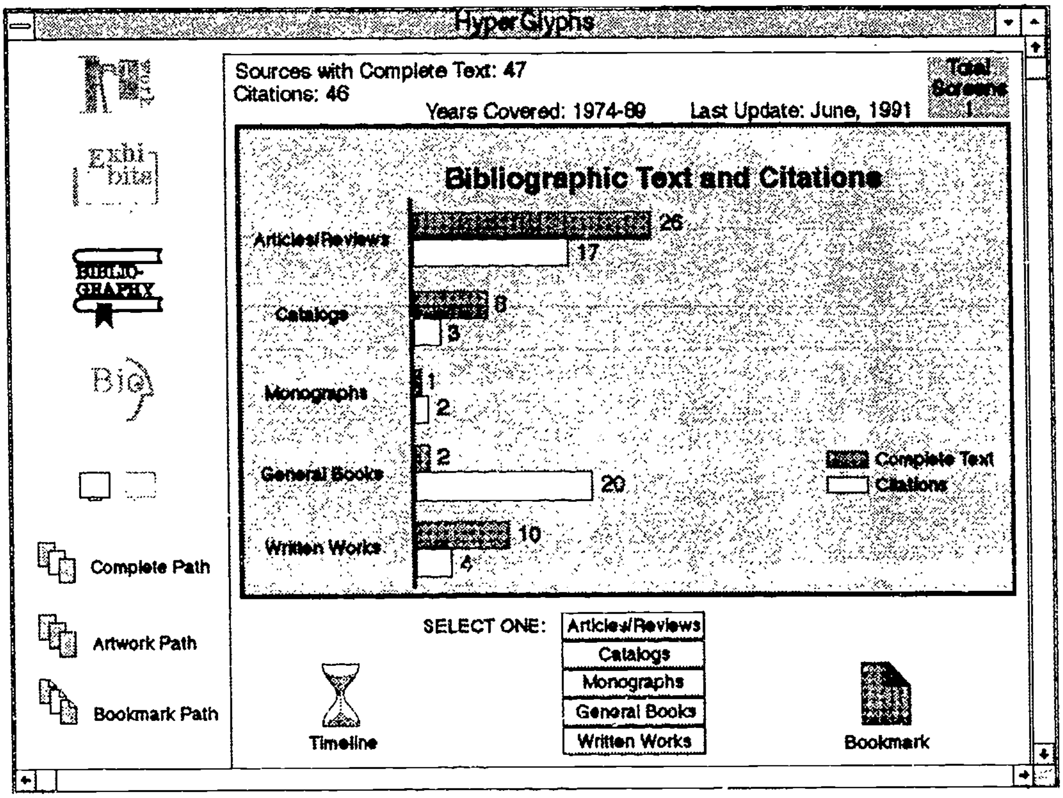


Figure 1. In the HyperGlyphs program, graphs provide overviews of the database. The program also indicates the period of time covered by the database and the date of the last update.

Some hypermedia interfaces use organizational charts to show all the predefined hierarchial and associative links in the database. The HyperGlyphs program does not use this visualization technique. The hypermedia interface should encourage the user to take an active role in defining a conceptual framework for navigation and information retrieval. Database diagrams that depict the predefined links in the program underscore the editor's perspective not the user's perception of the conceptual relationships in the database. Furthermore, although program designers can define and visualize hierarchial relationships in interactive programs, they have not developed effective methods of tracking and visualizing associative links (Hara, Keller, & Wiederhold, 1991). As a result, there is a lack of design guidelines for specifying associative links in hypermedia user interfaces.

In addition to providing overviews of the database to facilitate navigation, a hypermedia interface should provide spatiotemporal cues for the individual references. These cues help the user conduct a visual analysis of a document to determine its length and contents. Spatiotemporal cues also establish landmarks for information retrieval. Research has shown that orientation and recall are linked to spatial location within a body of text (Lovelace & Southall, 1983). In the HyperGlyphs program, a header at the top of each screen tells the user the number of illustrations in the document and the length of the document in terms of "computer screens." As the user scrolls through the reference, dashed lines with numbers indicate the divisions between screens and establish orientation cues for navigation and information retrieval (see Figure 2).

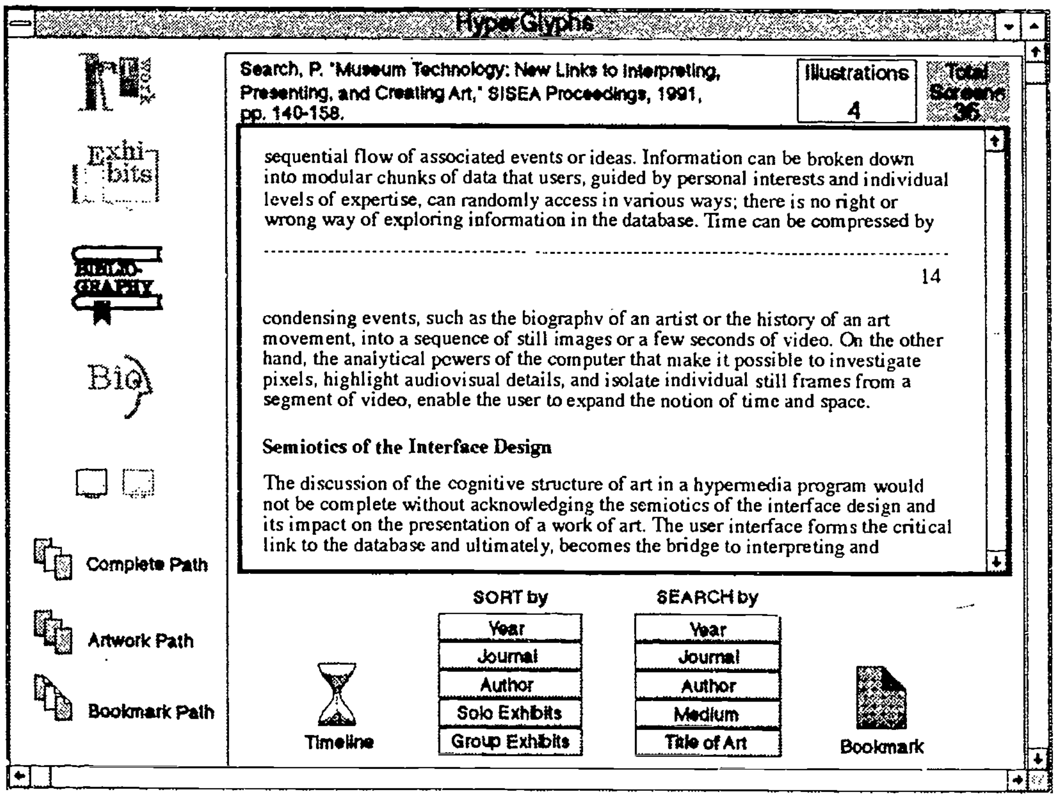


Figure 2. The HyperGlyphs program provides information about the content and size of documents. Dashed lines and "screen" numbers within the text establish navigational cues.

Narrow the Objectives of Navigation Functions

Hypermedia programs can improve navigation by providing users with a record of the information that they have already seen (Simpson, 1989). Many hypermedia programs provide navigation functions that track the path of the user. However, these functions are usually multipurpose functions that attempt to accomplish the following objectives:

- Allow the user to backtrack and retrieve information;
- Locate the user's position in the database;
- Identify what information has been accessed and what additional information is available;
- Show the hierarchial and associative links in the database (Search, 1993a).

These navigation paths are often visualized as a tree-structure diagrams that show the current location of the user in the database and the hierarchial and associative relationships in the database structure. Unfortunately, this type of "You are here" approach to depicting the user's path creates confusion because it tries to fulfill too many objectives. If the purpose of the navigation function is to allow the user to backtrack and retrieve information, the path does not have to show all the remaining information in the database or the hierarchial and associative links to the references. These conceptual relationships can be depicted by other functions in the program such as functions that generate textual or graphical overviews of the database (Search, 1993a).

The HyperGlyphs program limits the objectives of the navigation function to backtracking and information retrieval. The program keeps an internal record of all the data that the user

BEST COPY AVAILABLE

ERIC

accesses during a given session and then displays a list of that information upon request. This navigation list consists of textual rather than pictorial references to the data. By using a textual list, it is possible to display long navigation paths on any type of computer screen regardless of its size or resolution. A linear list is also an accurate visualization of the sequential *process* of navigation. While it is true that hypermedia programs support non-linear, random access to information, the navigation path itself represents a serial route through the database because only one reference is accessed at a time (Search, 1993a).

Conclusion

Hypermedia programs must provide orientation references that help users navigate through different levels of authorship and complex arrays of multimedia data. The navigational protocols of traditional hardcopy are not adequate for these dynamic, user-defined information structures.

The design of hypermedia navigational aids must begin with a critical analysis of the psychodynamics of hypermedia computing. The interface should anticipate the user's perceptions and expectations and help the user realistically evaluate the technology's potential for accomplishing a set of tasks. The user can then define a cognitive approach to information retrieval and become an active rather than a passive participant in the navigation process.

The conceptual design of a hypermedia program must be carefully mapped to the user's tasks and objectives. This process determines the organization of the database and the types of interactive functions. Translating this conceptual model into a user interface requires a critical analysis of the semiotics of the interface design. Design elements such as color, typography, layout, and language can establish important orientation cues for navigation. Since is possible to integrate layers of meaning into graphics and text, this type of relational coding is an efficient way to define the semantic structure of the program, the hierarchial and associative links in the database, and the syntax of the program functions.

In the future, the semiotics of the interface design will take on new dimensions as global networks develop multimedia interfaces. Semantic-syntactic coding in these interfaces will continue to play a critical role in navigation and information retrieval. However, the audiovisual cues in these interactive networks will assume another important function. In these networks, it will be possible to integrate cultural references into the audiovisual symbols and in turn, create interface designs that establish perceptual and cognitive bridges between cultures (Search, 1993b). These hypermedia interfaces will expand the creative potential of telematic communication by defining a new multicultural language that preserves and articulates diverse cultural perspectives within the context of an integrated global village.

References

Arons, B. (1991). Hyperspeech: Navigating in speech-only hypermedia. In *Hypertext '91 Proceedings.* New York: ACM Press.

Barnard, P., Morton, J., Long, J., & Ottley, P. (1977). Planning menus for display: Some effects of the structure and content. In *IEEE Conference Publication #150.* New York: IEEE.

Hara, Y., Keller, A., & Wiederhold, G. (1991). Implementing hypertext database relationships through aggregations and exceptions. In *Hypertext '91 Proceedings.* New York: ACM Press.

Landow, G. (1989). Hypertext in literary education, criticism, and scholarship. *Computers and the Humanities*, 23(3), 173-198.

Lewis, C., Polson, P., Wharton, C., & Rieman, J. (1990). Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. In *CHI '90 Conference Proceedings.* New York: ACM Press.

Lovelace, E.A. & Southall, S.D. (1983). Memory for words in prose and their locations on the page. *Memory and Cognition*, 11(5), 429-434.

Mullins, P. (1991). New images of text, text-making and text-reading. *Humanities Education*, 8(1), 4-10.

Search, P. (1992). Contemporary art: Today's window on tomorrow's communication media. In J. Clark-Baca, D.G. Beauchamp, and R.A. Braden (Eds.), *Visual Communication: Bridging Across Cultures*. Blacksburg, VA: International Visual Literacy Association.

Search, P. (1993). The art and science of hypermedia. In R.A. Braden, J. Clark-Baca, D.G. Beauchamp (Eds.), *Art, Science & Visual Literacy*. Blacksburg, VA: International Visual Literacy Association.

Search, P. (1993). The rhythm and structure of multicultural communication. In *Proceedings of the Third International Symposium on Electronic Art (TISEA)*. Sydney, Australia: TISEA.

Simpson, A. (1989). Navigation in hypertext: design issues. Paper presented at International OnLine 89 Conference, London, England.

Snowberry, K., Parkinson, S., & Sisson, N. (1985). Effects of help fields on navigating through hierarchial menu structures. *International Journal of Man-Machine Studies*, 22, 479-491.

Wright, P. (1991). Cognitive overheads and prostheses: Some issues in evaluating hypertexts. In *Hypertext '91 Proceedings*. New York: ACM Press.

Zellweger, P.T. (1989). Scripted documents: A hypermedia path mechanism. In *Hypertext '89 Proceedings*. New York: ACM Press.

Zuboff, S. (1988). *In the age of the smart machine*. New York: Basic Books.

Acknowledgements

I would like to thank Adelaide Lane, Manager of Instructional Computing in Rensselaer's Engineering Computing Services, for providing technical support for the development of the prototype program. I would also like to thank all the students who wrote software for the project.

HyperGlyphs Screen Designs. Copyright 1993 by Patricia Search. All rights reserved.

Education by Engagement and Construction: Experiences in the AT&T Teaching Theater

BEN SHNEIDERMAN
Department of Computer Science
Human-Computer Interaction Laboratory & Institute for Systems Research
University of Maryland, College Park, MD 20742

Introduction

Novel technologies can stimulate innovative solutions to traditional problems. During the past two years, I have been challenged to revise my notions of learning because I have been teaching in the AT&T Teaching Theater at the University of Maryland. This unique classroom was designed to enable instructors and students to explore new strategies, by supporting extensive use of media, computing, and network supported collaboration. I have learned many lessons about technology and education in the AT&T Teaching Theater, but my most compelling experience is the acceleration of my transformation as a teacher. While I appreciated the wisdom of and was influenced by many great educators such as Dewey, Bruner, Piaget, and Papert, and the reports of national committees [NIE84, AAH87], I have made my own synthesis [Shn92a, b, c]. I see more clearly than before that the path to motivating students is the joy of creation, exploration, and discovery. I see also that these processes are social in nature and that shared experiences in class and through teamwork projects are vital.

I have tried to describe this philosophy of education with the two-level phrase "engagement and construction." At the classroom level, students are engaged with each other in an active way to construct something meaningful and substantial. They work in teams to write a report, produce software, make a presentation, prepare a videotape, etc. At the community level, students are engaged in the real world, relating to "clients" (professionals, parents, children, patients, etc.) to construct something of value and importance to them. The team projects might be to computerize a list of hospital volunteers, propose a new ethics policy for the university computer center, develop a hypermedia guide to computer virus protection, write a handbook on software aids for deaf children, etc.

I claim that computing and communication technologies have opened up new possibilities in the educational environment by empowering students in remarkable ways. Students can now apply their skills to socially valuable projects, benefiting their clients while gaining meaningful knowledge plus having satisfying experiences. This approach is quite different from many views of computing and education that still depend on the archaic fantasies of teaching machines, computer-assisted instruction, intelligent computer-assisted instruction, and intelligent tutoring systems. These terms emphasize the machine and the delivery of instruction as if knowledge could be sent by modem at 9600 baud. These systems often make the student the victim of the machine. Empirical studies show limited or moderate benefits to these approaches and my experience is that the main beneficiaries of an educational experience are those who create the systems. The newer view focuses on interactive learning environments, discovery learning, and constructivism. Students seem likely to learn more by having to create an instructional module or a hypermedia presentation than to use one. Similarly viewing a videotape can be instructive, but creating one compels a student to dig deeper, become more engaged, and gain a far greater sense of accomplishment. In many educational circles, evaluation is shifting from mind-numbing and fragmenting multiple choice questions to holistic processes and project oriented methods. Instructors and students may be cautious about departing from familiar patterns, but with careful first steps I believe that most courses can be gracefully and evolutionarily restructured. The basic tools of creation are wordprocessors, paint programs, spreadsheets, database managers, communications packages, and programming languages. Of course, some of the ideas I propose could be done with low or no technology, but these novel powerful tools empower students in remarkable ways. I hope that my experiences can help foster confidence, inspire some teachers, and stimulate others to improve on my methods.

Technology to Support Learning

The concept paper for the AT&T Teaching Theater was written in 1988 by the Steering Committee at the University of Maryland. We followed the advice from colleagues at other universities (especially the University of Arizona) who had built electronic meeting rooms for business-related group decision support systems, but our goal was to create a learning environment for undergraduate and graduate education. Our plan was to support instructors in every discipline, not just computing, business, or engineering, and to limit use by outside companies or government agencies. Although we have been strongly tempted to seek income from rental to outsiders, we have largely stuck to our principles. Our vision was to create a new environment that makes ambitious use of media, provides computers for students and instructors, and supports collaboration by networking.

The media components include VHS and UMatic tape players, audio tape, CD players, laser disc, 35mm slide projection, and overhead projector. Two large screen high resolution projectors (768 x 1024) can display the media or the student and instructor computers. There is a touchscreen control panel for media and lighting, but its complexity is unsettling even to long-term users such as myself. There is a white board between the two projectors.

The computer components are 20 student and 1 instructor AT&T 386PCs with Super VGA monitors, a mouse, and a keyboard. The computers are housed in an adjoining room for security, to reduce noise, to minimize heat, and to reduce the clutter in the classroom. We installed the monitors down in custom built desks to preserve sight lines and eye contact. These strategies have worked out well and the classroom is an appealing environment for teaching, even when the computers are not in use. Reduced desk space and limited leg room are modest disadvantages. A tiered floor elevates the rear rows, but wheelchair access is provided to the first row of five computers The twenty computers were intended to be shared and therefore there are 40 chairs. Careful installation of lighting, air conditioning, carpeting, and wall treatments, plus comfortable chairs have made the room attractive for students and visitors. After much debate we chose to use Microsoft Windows 3.1, and many instructor requested packages (e.g. Quattro Pro, Paradox, MS Write, Visual Basic) are mounted as needed.

The networking is provided by a Novell Local Area Network, which is connected to the campus network. Within the classroom shared access is quite good, but there are limitations when outside the classroom. On campus users who go to computer labs equipped with the Novell LAN and Windows 3.1 can access the machines in the Teaching Theater, but other access is limited to file transfers. Students and instructors were pleased when it recently became possible to access files from UNIX workstations and by dial-up from off-campus. The desire for connectivity is strong.

A further support for collaboration is an electronic switch box that enables instructors to select any student machine and view it, display it on either large screen projector, or show it on all 21

ERIC

computer monitors. This is a very rapid and easy to use tool that I depend on heavily. There is an effective support staff that maintains the hardware, installs software packages, develops novel software, and assists instructors and students. The room is monitored with a team of part-time undergraduate and graduate students, so that instructors can call on help with the computers, media, lighting, etc. during every class.

Scenes from the AT&T Teaching Theater

The technology is only the stage on which to produce educational experiences, and the creative work of numerous instructors is daily yielding novel scenarios. I will report on my favorite moments and appealing ideas from other instructors. Some days are less exciting and I may not use much of the technology, but I have become less apologetic about these quieter days. It takes preparation to make great and innovative things happen, but over time I have become more familiar with the room and more spontaneous. The novelty and preparation are significant barriers for many instructors who have well-developed course plans and lecture notes. Even for technology-oriented instructors, it is a challenge to restructure their courses and adapt their teaching styles to take advantage of the AT&T Teaching Theater.

Classroom Engagement and Construction

Instructor's blackboard: The simplest and most common scenario is for me to use the wordprocessor on the instructor's workstation as my blackboard. In some courses, I begin each class by putting up the course outline (in a large font) to see our progress, note when assignments are due, and remind students about readings. Then I will switch to a text file with notes from the previous class and we can review our discussion, refresh our memory, and give a quick overview for absent students. If appropriate, I will extend the notes file with today's discussion items and at the end of the class will copy the file to the shared directory so that all students can review the notes. I like asking students a question and then typing in their answers, refining the answers, organizing them and winding up with a coherent set of notes. I might be able to prepare slides with my answers in advance and hand them out, but the process of creating the notes communally seems to engage students in a lively discussion. As in most classes, some students are more vocal, so I try to ask for contributions from as many students as possible. I have also used the Paintbrush program to make sketches in class. My early drawings were clumsy but I have gotten smoother and know better what works quickly in front of a classroom. Even 5-15 seconds of my fumbling through the menus to find what I want is irritating to the students and embarrassing to me. Similarly, when I'm using a new database package or other tool, I find it important to rehearse me lectures and make sure I know the commands I need.

Student presentations: I keep reminding myself to give my students a chance to use the room and to see what they come up with. One strategy in my senior database class was to have a pair of students volunteer to go over the solution to each week's homework problems. They came to the front and used the instructor's workstation to show their solutions on the large screens. This worked well in that the presenting students worked especially carefully on their solution set and got a chance to show their best. Often students in the class would offer other solutions and we could switch to their machines to see what they had come up with. I also arranged for student presentation of term projects, sometimes including videotapes produced with the camcorder I lent them.

Class exercises: In my user interface design course. I ask students to work in pairs to design pieces of a user interfaces such as a menu hierarchy, a form fill-in, or a set of icons. They have gotten quite lively in their discussions and more confident about presenting what they create. I will give them 15-30 minutes and then use the switch box to show all the versions quickly. I linger on the designs I consider excellent, thereby reinforcing high expectations and allowing students to see how well their colleagues can do. In the icon design exercise one of the students made a playful but disbelieving remark about what one of the teams accomplished. I think everyone was impressed and got some fresh ideas about what was possible. The public embarrassment for the weaker students is of concern to me.

Homework projects: Teamwork design projects accomplished over a three-week period were also part of my user interface course. Students had to design a mockup of a video-on-demand system that would enable home viewers to select one out of 10,000 videos. One of the students downloaded a database of movie reviews and made it available to the others so that realistic titles, reviews, and other information could be shown. Students worked in Visual Basic and then showed their mockups to the class for a critiquing session that lasted almost an hour.

Group programming: One of my great thrills was teaching modular design in Pascal to first semester students. I composed a 20 line main program that invoked three procedures. I asked students to generate versions of the procedures and then examined what they did using the switch box. I chose the one I liked best, made a few slight changes, and then copied it from their machine to mine. I pasted together the components and was thrilled when it compiled and ran on the first try. While I was enthusiastic, my students seemed to think that this was the natural way to do things.

Course reviews: We use a groupware system, Visionquest, to enable students to comment on the course in topics such as the homework, projects, tests, lectures, textbook. and the classroom. Students select a topic and type brief comments which are shown on the large screen anonymously. In 15 minutes I have gotten 168 one- or two-line comments. I find it fascinating so see my student's comments, although it is somewhat painful to see my weaknesses clearly identified. I review the printout and give my feedback on their comments at the next class session. This year students vigorously complained that my first take home test was too long, although others defended its fairness and utility as a learning experience. The next test will be shorter.

Brainstorming: My colleague in the College of Business, Professor Maryam Alavi makes heavy use of Visionquest in her case-oriented Management Information Systems course. She might ask her students to comment on ways for a stereotypic company to increase sales or control costs. The suggestions can be viewed, organized, ranked, rated, and refined. She teaches in the evening and reports that, while in other courses students are often tired and eager to leave, her students often stay late in the Teaching Theater to add detailed comments for her case studies.

Hypercourseware: Another colleague, Professor Kent Norman of the Department of Psychology, has invested a substantial effort to develop extensive notes for his course on Experimental Statistics [Nor90, Nor93]. He uses Plus (from Spinnaker Software) to build slides with step-by-step demonstrations of statistical tests and sample data for students to manipulate. These are integrated into a full course support system that manages the student roster, presents the course outline, offers background readings and references, etc. all linked together. Hypercourseware has been used as the platform for three courses—Statistics for the Behavioral Sciences; Cognitive Psychology: Thinking and Problem Solving; and the Cognitive Seminar. The point of Hypercourseware is to build from an integrated set of stacks that represent traditional objects of classroom education: the syllabus linked to lectures and exams, the class roll linked to biographical sketches and pictures, and the class seating chart. In statistics the fun is group data collection, dissemination, and analysis. In thinking and problem solving the fun is collaborative sharing of notes and ideas, brainstorming, and simulated cognitive experiments.

Community Engagement and Construction

My orientation towards engagement and construction in the classroom seems to lead quite naturally to engagement and construction in the wider community. While some students struggle with this new vision of education and resist my encouragement to go off campus, I do have enough successes to keep me exploring refinements to these strategies. Dealing with a dozen 3-person teams is demanding for an instructor, but I become engaged in the process and am eager to know more about what has been happening in each team.

Sharing computing skills: My favorite story from a Computers and Society course is how a team of students interested in computers for the elderly worked with the director of a nearby senior citizens center. After reading the background literature the team planned a set of instructional experiences for the seniors, exploring word processing paint programs, games, etc. The students validated their theory about what would work and what would not and produced a report for the director of the center about what they might take on as next steps. Another team of students worked with a local junior high school to reorganize their computer lab and develop a plan to better support the schools educational mission.

On campus database projects: In my database course students took on more software development projects, such as helping the journalism department keep a list of more than 800 internships available to students nationally. Another group built a complete accounting package for the veterinary sciences department, while a third kept track of software site licenses for the computer center. These and other projects produced software that continues to be used. A less successful but more ambitious project was to attempt to arrange carpools on campus. They used the address information in the faculty-staff directory and struggled to find an effective geographic matching mechanism. The goal was to send e-mail messages to potential car pool partners and encourage them to make connections. The team built some interesting neighborhood finding algorithms based on zip codes, but it was not adequate for the task. The Office of Commuter Affairs was interested in the project and maybe a future team project can build on this initial effort. The team still got an A for a thorough and thoughtful report on their analyses. Students could demonstrate their projects to their classmates easily in the AT&T Teaching Theater.

Off campus database projects: My students were reluctant to make contact with potential off campus clients but several successful projects emerged. One team built a database to store information on 23,000 donors for a major charitable organization, while another team kept track of thousands of attendees for Montgomery County community center courses. Both continue to be used. Many students have off campus connections through their family or part-time jobs, and where possible I steer students to clients that I have identified through my contacts. I will be even more devoted to off campus projects that provide community service to soup kitchens, health hotlines, clinics, and legal services organizations. I believe undergraduates can provide meaningful assistance to these volunteer organizations.

Graduate Seminars: A typical approach to a graduate seminar is to read a research journal paper each week and have one or two students start the discussion by critiquing the paper. In my experience, students will often make harsh and sometimes unfair criticisms of the paper so as to demonstrate their cleverness or insight. Using the principles of engagement and construction as a guide, I now require my graduate students to compose a letter to the authors and send it to them by electronic mail. This requirement has made discussions much more thoughtful and evoked many more positive statements about each paper. My students' comments have been well-received and yielded appreciative comments. In one case, a student was acknowledged in a subsequent paper by the author.

Empirical studies research projects: One of my great pleasures is to have undergraduates work on empirical studies research projects and then publish papers with them as co-authors. I would say that one in ten such project yields publishable results and then there is often much additional work to make a professionally polished paper. I believe that this expectation is realistic in many areas if the instructor is willing to work with students and closely monitor their efforts. In my course on Human Factors in Computer and Information Systems I have seven milestones with deliverables before the complete project is submitted: proposal, initial materials, bibliography, pilot test results, initial statistical analyses, raw data reports, and first draft of the introduction.

The students take great pride in their accomplishments and learn a lot by hearing about other projects and being pilot test subjects for their colleagues. End-of-semester presentations are also an important experience for both the presenters and the listeners.

Conclusion

These examples are an informal report about my initial experiences with a new teaching philosophy and a new technology. Maybe the lessons are separable but there seemed to be a synergy between my emerging principles and the technology. There are still many problems for which I seek better solutions such as how to grade students in teams, how to bring these principles to larger courses, and how to ensure fulfillment of traditional course requirements.

Next semester I will teach a new graduate seminar on Virtual Reality, Telepresence and Beyond, but this will be by satellite television and it will be outside the AT&T Teaching Theater. However, the lessons I have learned are guiding me to using electronic mail, team projects, and high expectations for my students to contribute in a meaningful way to the current research.

Not all teachers have facilities such as the AT&T Teaching Theater or satellite television, but access to computing and networking is quite broad and improving. Administrators should be encouraged to support development of these facilities. Education by engagement and construction applies not just to computer science courses but to every discipline and at every age, from elementary school to continuing education (Figure 1). The Final Report of the Study Group on the Conditions of Excellence in American Higher Education, National Institute of Education wrote that "Active modes of teaching require that students be inquirers—creators, as well as receivers of knowledge." That report also stressed projects, internships, discussion groups, collaborations, simulations, and presentations (Figure 2). Similarly, the Principles for Good Practice in Undergraduate Education presented by the American Association for Higher Education (Figure 3) pushed for cooperation among students and active learning projects.

Students want to engage with people to:

Create	Communicate	Plan	Help	Initiate
Explore	Build	Discover	Participate	Collaborate

Students will be engaged by constructing products:

Writing (poems, plays, essays, novels, newspapers, diaries)
Drawing (pictures, logos, portraits, maps, birthday cards)
Composing (music, songs, operas, hypermedias, videos)
Designing (buildings, furniture, games, animations, family trees)
Planning (class trips, charity events, vacations, parties, elections)

Teachers should promote:

Engaging in the world (lobby a Senator, raise environmental awareness, call City Hall to report a problem)
Helping where needed (teach computing to the elderly, improve recycling, increase awareness of drug abuse or AIDS)
Caring for others (raise funds for the homeless, improve medical care)
Communicating ideas (write to a newspaper editor, make a class speech, produce a cable TV show)
Organizing events (prepare a bake sale or lecture series)

Multimedia technologies can empower students:

Enable students to create multimedia reports
Encourage media-supported class presentations
Develop communication through electronic mail
Provide experience in searching databases
Explore information networks and bulletin board systems
Promote use of word processing, drawing, spreadsheets, . . .
Project orientation enhances engagement
Help an elementary school to improve computer use
Teach elderly users word processing
Find or develop aids for a handicapped person
Revise university policy on information protection and privacy
Improve university administration, registration, . . .
Evaluate and suggest improvement to bank machines, library systems, public access terminals, voicemail, . . .
Write guide for parents about kids' software
Review workplace practices for computer users

Figure 1. Strategies for increasing Engagement and Construction

(1) Student Involvement
- involving students in faculty research projects
- encouraging internships
- organizing small discussion groups
- requiring in-class presentations and debates
- developing simulations
- creating opportunities for individual learning projects

(2) High Expectations
(3) Assessment and Feedback

Figure 2. Conditions for Excellence in Undergraduate Education

Involvement in Learning: Realizing the Potential of American Higher Education Final Report of the Study Group on the Conditions of Excellence in America Higher Education [NIE84]

Encourage Student-Faculty Contact
Encourage Cooperation Among Students
Encourage Active Learning
Give Prompt Feedback
Emphasize Time on Task
Communicate High Expectations
Respect Diverse Talents and Ways of Learning

Figure 3. Principles for Good Practice in Undergraduate Education [AAH87]

References

[AAH87] American Association for Higher Education, 1987. Principles for Good Practice in Undergraduate Education.

[NIE84] National Institute of Education, 1984. Involvement in Learning: Realizing the Potential of American Higher Education, Final Report of the Study Group on the Conditions of Excellence in American Higher Education.

[Nor90] Norman, Kent L., 1990. The Electronic Teaching Theater: Interactive hypermedia and mental models in the classroom. Current Psychology: Research and Reviews, 9(2) , 141-161.

[Nor93] Norman, Kent L. & Lindwarm, Diane, 1993. Human/computer interaction in the electronic classroom. Proceedings of the Mid- Atlantic Human Factors Conference, Virginia Beach, February, 217-223.

[Shn92a] Shneiderman, Ben, 1992. Education by Engagement and Construction: A Strategic Education Initiative for a Multimedia Renewal of American Education, In, Barrett, Ed (Editor), Sociomedia: Hypermedia, Multimedia and the Social Creation of Knowledge, MIT Press, Cambridge, MA, 13-26.

[Shn92b] Shneiderman, Ben, 1992. Designing the User Interface: Strategies for Effective Human-Computer Interaction, Second Edition, Addison-Wesley Publ. Co., Reading, MA.

[Shn92c] Ben Shneiderman, 1992. Engagement and Construction: Educational strategies for the post-TV era, In Tomek, Ivan (Editor), Proceedings of the International Conference on Computers and Learning, Springer-Verlag, New York. Reprinted in the Journal of Computing in Higher Education, Spring 1993.

Acknowledgements: AT&T's generous funding made this educational exploration possible. All of us who participated in the process greatly appreciate their support. Ron Weissman and Glen Ricart played key roles in the development of the AT&T Teaching Theater plans. Walt Gilbert masterminded the development and construction efforts and Ellen Yu has done an excellent job in supporting teachers and students on a day-to-day basis. Constructive comments on early drafts of this paper were provided by Gary Marchionini, Kent Norman, and Ellen Yu.

ERIC

A Computer-Mediated Learning Environment for Adult Learners: Supporting Collaboration and Self-Direction

CHRISTINE STEEPLES
CSALT, Department of Educational Research
Lancaster University, Lancaster LA1 4YL, UK

This paper explores two key issues for computer-mediated learning environments; learner collaboration and learner independence and their relationship to the notion of public and private spaces within electronic learning environments. The significance in computer-mediated learning environments of human-human interaction and collaboration is being increasingly recognised, but there is a potential tension with well-established educational principles of self-direction and independence for adult learners. These themes are developed by considering work of the JITOL project: a project that is looking at the development of computer-mediated learning environments to support geographically dispersed groups of learners in their professional development. For the first user trials in the project, a communal hypermedia database system is under development. This system offers users the facility to operate in both communal, publicly accessible spaces and within private, individual spaces. The paper discusses how computer-mediated systems may support the professional development of adults by providing environments which allows individual needs and purposes to be met *and* experience and understanding to be shared.

How can we best support adult learners in their professional development?

This paper is concerned with the professional development of very specific groups of adult learners. We have gained experience of supporting one specific group of professionals through an advanced level programme of study at Lancaster University (MSc Information Technology & Learning). From this experience we are committed to a view that individual, adult learners need to be able to assume control (or certainly degrees of control) over their learning activities. This can be in terms of not only defining learning goals and tasks but also in terms of the timing of the activity and the location (Goodyear & Steeples, 1992).

Learner control and independence are key aspects in making learning meaningful for adult learners. Another aspect is the rich resource of each learner's unique experiences and differing contexts. If we are to draw upon the strength and potential of individual experiences, there is an argument for a blending of learning and ongoing activities. This allows context and experience to be utilised in a way that is meaningful to that individual's professional development.

Supporting professional development from a perspective of individual learner empowerment may suggest potential conflict with a perspective that promotes collaboration and sharing in learning amongst professionals within a specific field of interest. Yet these two themes are central to the environment that the JITOL project is seeking to develop.

The next section in this paper begins an exploration of the notion of collaborative learning. I then go on to outline the JITOL environment before a consideration of how technological tools may facilitate learning through collaboration amongst dispersed professionals.

ERIC

What motivates learners to collaborate?

Collaboration in learning activities has been described as offering a number of benefits (Harasim, 1990, Brown & Palincsar, 1989, Scardamalia, 1992, O'Malley & Scanlon, 1989). We are keen to explore these benefits more deeply in respect of distributed professionals. Briefly, the salient points are identified are as follows:

1) Most importantly, collaboration supports and encourages the sharing of individual expertise and experience.

2) Articulation of an individual viewpoint and the responding comments from others with differing perspectives can lead to refinement and deeper understanding, or higher level concept attainment. Articulation is also seen as a process that promotes the development of meta-cognitive skills such as concept refinement and revision.

3) Collaboration promotes the active construction of knowledge and engagement in the learning process.

4) Collaborative activity can lead to a sense of involvement and identification with a group and its product(s); this supports the human/sociological need to operate and interact with others.

5) Collaborative learning focuses on learners as active participants.

6) The articulation and refinement process can lead to a generalisation of the underlying principles which enables transfer to other contexts and broader application of the knowledge.

These benefits presuppose a view that knowledge should not be regarded as a private commodity. In fact I would emphasise that for the specific professional groups we are supporting in JITOL, the knowledge is inextricably bound between the professionals themselves and the practices they perform as part of their work. There is a constructive interplay between the knowledge, the professional (and amongst the professionals) *and* the practice. Sharing of contextualised experiences allows a natural way for these elements to be brought into play.

Collaborative learning has been closely associated with, and draws support from the research of Vygotsky upon the notion of a *zone of proximal development (ZPD)* (see Brown & Palincsar, 1989). There is a close association between research in ZPD and theories of constructive learning, cognitive apprenticeship, expert scaffolding and latterly situated learning in communities of practice (Lave & Wenger, 1991). The common thread to the discussions is to suggest that learning can be supported but remain under learner-direction by providing an environment (ie supports or scaffolds) that encourages the learner to extend and develop from their present position (their present understanding) to one of new knowledge and new understanding. The relevance to this discussion here on computer-mediated learning environments is that the scaffolds, the other participants and the thinking processes they display in a community, become accessible through the technological tools that such an environment can provide.

It is important to be clear in our definition of collaborative activity. Rather than groups formed to address a common task, JITOL learners may be exploring an area of common interest but for each, from an individual perspective and need. The environment provides a framework for participation but one that is learner-controlled. The primary focus then, is not to achieve a group objective, but to satisfy individual goals, calling as the need arises upon the collective effort and support of others.

The JITOL environment

In the JITOL project we are attempting to provide, through a technological base, a rich and supportive environment for human-human interaction. JITOL (or "Just in Time Open Learning) is a large, European project involving partners from large multinational companies, courseware production companies and educational institutions. The project began in January 1992 and will run for 3 years.

The JITOL environment is a mixing of electronic communications with domain-specific online information resources to support professionals in their work. The conception of a JITOL environment includes these professionals subscribing to a JITOL service tailored to their specific needs which provides the resources (both human and material) for them to access, at a time and in ways appropriate to their specific needs (the "just in time" part of JITOL). Access to JITOL is envisaged as a continuing activity, involving the setting of goals and interaction with other professionals within a practitioner community, particularly tutors or mentors and the exploitation of JITOL resources.

A description of JITOL learners

JITOL learners are drawn from three distinct areas:

1. Advanced learning technology professionals, (ALTP). This group is broadly spread, based within the training units of large national and multinational companies, in the public sector (such as the Health Service), in courseware production companies or in education both at tertiary and school level. Their unifying interest is in the design, production and use of IT-based learning materials (technology-based training). This paper particularly draws upon experiences in supporting the ALTP domain.

2. Doctors concerned with diabetics self-help therapy, (DSHT). These medical doctors are dispersed across Europe but share a common interest in advancing patient education- in the proactive involvement of patients in their own treatment and control of their diabetes.

3. CS or Corporate Staff. This group is made up of staff in a large financial institution (IFCAM, Credit Agricole, France) and engineers working for a high-tech multinational concern (Digital Europe).

Whilst the 3 areas or domains are widely different, and this gives us a broad range of opportunities to test and evaluate JITOL services, the common thread is support to distributed adults through the use of electronically-mediated communications.

The JITOL learners accord with research on the characteristics of adult learners (Brookfield, 1986, Cross, 1984). JITOL learners are highly professional, individual, adult learners, with assumptions about what they expect to gain from participation in learning activities. They are able and want to assume higher levels of control and responsibility over their learning activity and perceive themselves as self-directing adults.

In the ALTP domain particularly, the learners have experiences of learning in practical situations; they have unique models and assumptions of how learning takes place. Many work in relative isolation (ie. from other learning materials designers) and have, by the nature of their working practices, continually to assume control and direction. Knowledge and expertise in this domain is new and non-formalised. Much of the knowledge resides within the individuals in this community, tied to their experiences, perspectives and differing situational contexts. Learning activities need to be tailored to fit into this mix of knowledge and practice if they are to be as rich and meaningful as possible.

In the domain concerned with diabetes therapy (DSHT) there is no "set", prescribed course of study. The doctors represent a group of like-minded, but geographically distributed, individuals- a practitioner community. Within this community, there are individual learning goals which each participant can identify and define through support from their colleagues. They are entirely self-directed and responsible for identifying their own needs but can appreciate and draw strength from the mutuality. They share and exchange ideas and individual expertise for the common good.

Support of learner/peer and learner/tutor relationships through computer mediated communication

We are using asynchronous computer-mediated communications in the JITOL project. Computer mediated collaboration in this medium can be assessed along several dimensions. The first dimension to stress is *temporal*. Collaboration operated in an asynchronous mode supports self-pacing, reflection and idea formulation before response. This may well enhance the quality of the discussion; the purposeful construction of a response. Collaborative learning requires articulation to others which can aid the development and refinement of understanding. Learning may typically also involve an initial, internal, articulation or rehearsal of an idea or point of view. The process of rehearsal, articulation and refinement may become multi-layered through the electronic medium.

An environment that is electronically mediated allows establishment and maintenance of relationships over time (as well as space, see below) and weakens the restraints of time that may be imposed on other forms of group activity.

A second dimension is *geographic* because learners are no longer constrained by location limitations. Collaboration with peers and groups with a common interest can take place at a location to suit each individual, irrespective of distance. More positively, each participant is allowed direct access to the central focus of the work, task, or discussion irrespective of distance. Since access to the JITOL system will be the primary activity for our learners, it is important that this does not exacerbate feelings of isolation and remoteness, so often associated with distance study. The formation of groups to actively participate in collaborative learning activities can diminish isolation.

Another dimension to consider relates to *human-sociological* factors. Evidence exists to uphold that electronically mediated communications can diminish social barriers to group communication such as that of individual status. Dubrovsky, Kiesler & Sethner (1991) talk of the "equalization phenomenon" that computer mediated communications can offer in this respect. Electronic messages are stripped of visual and non-verbal cues, enabling the reader to concentrate upon the cognitive content, it has been argued. The learner may draw upon the support and guidance of more expert participants with less concern to their own individual status as perceived by others in the community.

It may also be argued that the proximity to more "knowledgeable" others that is offered through this type of electronic environment may be beneficial. A new learner may access the discussions of others. Scardamalia & Bereiter (1992) talk of "purposeful browsing".

Hypermedia Environments to Support Learning

The JITOL project is developing a communal hypermedia database system that will develop in response to learner activity and learner needs. I concentrate here upon the concept of *organic* growth to suggest the idea of hypermedia environments that can grow in a creative, responsive way thereby mirroring the growth and evolution of human intellect.

A strong assertion made about hypermedia environments is that they parallel human (cognitive) methods of organising information. Nelson and Palumbo (1992) discuss this equivalence, but pin-point various drawbacks or limitations of present hypermedia systems in contrast to the highly refined and multi-faceted sophistication of human memory - particularly its ability to make adaptive associations and convey relational information in the association. However Nelson and Palumbo do assert that a sub-strand of hypermedia environments, which they refer to as "knowledge construction environments", with tools to support user annotation and adaptive linking, allow the user to construct a dialogue with the system. By doing so the user is allowed self-direction, to make the connections he/she wants to make, in a powerful environment for supportive, constructive learning. Moreover the user is supported in adding his/her perspective to the knowledge-base. This is important to allow the natural and evolving re-orientation of the information to the actual, individual and current needs of the learner.

One example of a collaborative hypermedia environment that supports and encourages learner annotation is the Canadian system, CSILE (Scardamalia, 1989, Scardamalia & Bereiter, 1992). CSILE focuses on the collaborative development of a communal "knowledge space" . CSILE is a networked system allowing simultaneous access to a communal database of textual and graphical notes produced by children. Tools exist to allow the children to search through and comment upon each other's contributions.

CSILE has been a source of inspiration for development work in the JITOL project, not least because CSILE encourages the development of active learning processes whilst drawing upon the knowledge and expertise of individuals within the group.

How JITOL's hypermedia environment will support self-direction and collaboration

A Communal hypermedia database system

In the JITOL project we are aiming to produce a communal hypermedia environment that marries together tools and features of computer-mediated communications (email, computer conferencing etc.) with a hypermedia database. This facility is designed to support users in their access and exploitation of the JITOL resources (Albertsen, 1992). Additionally, we see the need to develop a system that facilitates both public, communal areas and private, individual areas as critical. This is to enable both individual reflection and collaborative exchange amongst the participants and includes facilities to annotate and make personal/ private associations between information in the database.

Briefly, the system consists of two parts that build upon an object-oriented database management system: a *Link Management* module (LM) and a *Note Editor* (NE). The LM module is a library for establishing and making use of links between objects of any class. Tl ; can be accessed from any JITOL application (word-processor, graphics package for example).

The Note Editor is a simple tool for managing notes (contributions of thoughts, ideas, discussions) and calls upon the LM to create links and associations between the notes and any other object class. The NE also contains tools and facilities for using links and indexes to search the objects in the database. Note format is deliberately simple, to ease and encourage use and participation in collaborative discussion. Notes may contain both textual and non-textual information. These functionalities blend well with our concerns to promote rapid-fire exchange of ideas in initial collaborative activity and goal definition activities.

The Note Editor facility has much of the functionality of a conferencing system; communal access to the information or notes, organised around a common area of interest. Notes may be collected into Note Sets around a topic or theme in the same way that conferences are organised. However there are essential additional features in this development that currently are not intrinsic to conferencing tools. These include sophisticated searching facilities and indexing and storage mechanisms to cope with an increasingly complex, permanent store of user contributions.

The JITOL communal hypermedia knowledge base allows the efforts of a group discussion around a topic or resource (including both online and offline resources) to be recorded and remain as a note set for each participant. Subsequently, other JITOL learners with similar concerns, may gain access to the information; to refine and reflect upon it for their own purposes.

Another key aspect of the JITOL environment will be to support both public and private "spaces". Private spaces may be restricted to one user or to a specific group. These private spaces are created by making specific associations between the notes. The note remains for public access - it is the unique collecting together (through association) that becomes individual. It is also possible to restructure these associations, make new associations and sets as the learner's knowledge changes. This facility allows a natural organisation, mirroring human (cognitive) ways of organising information. The environment is both organic and adaptive for each learner, enabling a progression from a supportive, to an increasingly complex and independent, structuring. There is provision too for learners to explore ideas before articulating to others.

Creation of private group areas in the hypermedia environment could be compared to the coffee bar discussions that take place between learners in traditional learning environments- separate from tutors, supporting peer discussion and debate to tease out the issues and problems. Whilst these may be created as part of a group process, it is possible for each learner then to impose their own order and arrangement upon the items, to delete (from their space) items they do not wish to include and to make connections (associations) to other notes, or note sets, in the database.

The developing hypermedia database aims to support and enrich exchanges. By being able to store, structure, search and reuse these discussions, we enable the learners to take even greater control of their learning and for the resources to develop in ways that suit the users' needs. Our experience has shown for example that there are similar questions that learners frequently ask within the ALTP community. By searching a structured store of previous discussions, we can allow the learners to examine other perspectives and create an organic environment that represents actual learner's situations, understanding and concerns.

How future JITOL developments and action will be assessed

First large-scale user trials began in October 1992. The user trials are seen as a vital component of the whole JITOL project- a positive force to promoting the project as an iterative, user requirements elicitation exercise. Early studies in the project work on user requirements have revealed uncertainty and hazy pictures of what the JITOL model could offer, and what it is realistic to expect from such services. Experience with these technologies will enable the JITOL users to more precisely articulate their requirements, enabling further needs elicitation and feeding into future developmental work.

ERIC

Running parallel to the large scale trials will be a number of focussed, small scale trials to test discrete components of the anticipated JITOL system functionality. (Goodyear, in press). These particularly concern developmental work on the communal hypermedia database described above and conducted within the ALT community of professionals.

Summary

My purpose here has been to look at some critical issues in the development of an environment appropriate for a specific class of learner. In particular I have been concerned that, on the one hand, research of adult learners reveals a strong need for individual learning activities, coupled with a need to be in control of their own learning. But, on the other hand, because of the knowledge and experience of each individual participant, the most powerful resources are these professionals themselves. The major share of the knowledge for their area of expertise resides in their own heads and tied to their practices. Learner collaboration not only emphasises a positive, constructive approach to learning but it also allows the knowledge and skills of the participants to be shared with their peers and with others who have similar interests and concerns. This process is further enriched because these collaborating professionals may well bring a whole pattern of experiences and different contextual concerns to the discussion.

Technological support imposes a new and dynamic dimension upon the collaborative learning process. Hypermedia learning environments can offer users the facilities to annotate and make unique associations; to add their own commentaries on resources and discussions that persist as the transcripts of real needs; a natural and evolving re-orientation of the learning materials in answer to specific needs (Goodyear & Steeples, 1992).

Hypermedia learning environments also allow the formation of communities of practice that are unhindered by time, location or status constraints. The learner can seek advice and support from fellow participants,whether learner, tutor or mentor, and increasingly move to more complex and engaged roles within the community. The barriers of traditional roles are diminished and individual expertise becomes available, to be shared in the community.

The JITOL project is concerned with how computers can be used to support a geographically dispersed group of learners in their professional development. It is concerned with pedagogical approaches that support learning that is both learner-centred *and* collaborative. These approaches need not be mutually exclusive. My discussion here has aimed to open this debate - developments and subsequent testing will reveal how supportive and appropriate the JITOL environment can be to these specific learners.

References

Albertsen, K. (1992) *Implementing a LAN-based hypertext system.* JITOL Deliverable 5 (Restricted), Workpackage 1.

Bannon, L.J. (1986) Computer mediated communication. In D.A. Norman & S.W. Draper (Eds.) *User centred systems design: New perspectives on human-computer interaction* (pp. 433-452). Hillsdale,NJ: Lawrence Erlbaum.

Bereiter, C. & Scardamalia, M. (1989) Intentional learning as a goal of instruction. In L.B. Resnick (Ed.) *Knowing, learning and instruction: Essays in honor of Robert Glaser* (pp.361-392). Hillsdale, NJ: Lawrence Erlbaum.

Bikson, T.K. & Eveland, J.D. (1990) The interplay of work group structures and computer support. In J. Galegher, R.E. Kraut, & C. Egido, C. (Eds.)*Intellectual teamwork: Social and technological foundations of cooperative work.* (pp.245-290)Hillsdale, NJ: Lawrence Erlbaum.

Brookfield, S.D. (1986) *Understanding and facilitating adult learning: A comprehensive analysis of principles and effective practices.* Milton Keynes: Open University.

Brown, A.L. & Palincsar, A.S. (1989) Guided, cooperative learning and individual knowledge acquisition. In L.B. Resnick (Ed.) *Knowing, learning and instruction: Essays in honor of Robert Glaser* (pp.393-452). Hillsdale, NJ: Lawrence Erlbaum.

Cross, P.K. (1981) *Adults as learners: Increasing participation and facilitating learning.*San Francisco, CA: Jossey Bass.

Dubrovsky, V.J., Kiesler, S. & Sethner, B.N. (1991) The equalization phenomenon: Status effects in computer-mediated and face-to-face decision-making groups *Human Computer Interaction 6*, 119-146.

Galegher, J., Kraut, R.E.. & Egido, C. (Eds.)(1990)*Intellectual teamwork: Social and technological foundations of cooperative work.* Hillsdale, NJ: Lawrence Erlbaum.

Goodyear, P. (forthcoming) Computer-supported collaborative learning in distance education. In M. Spector & D. Gettman (Eds.) *Research issues in distance learning* , Educational Technology Press. (Obtainable from the author as CSALT Working Paper 92/5)

Goodyear, P. & Steeples, C. (1992) IT based learning: Tasks and tools. *Journal of Computer Assisted Learning, 8* (3), 163-176

Harasim, L. (1990) Online education: An environment for collaboration and amplification. In L. Harasim (Ed.) *Online education: Perspectives on a new environment* New York: Praeger.

Lave, J. & Wenger, E. (1991) *Situated learning: legitimate peripheral participation.* Cambridge, UK: Cambridge University Press.

McCreary, E.K. (1990) Three behavioural models for computer mediated communications. In L. Harasim (Ed.) *Online education: Perspectives on a new environment* (pp. 117-130). New York:Praeger.

Nelson, W.A. & Palumbo, D.B. (1992) Learning , instruction and hypermedia. *Journal. of Educational Multimedia and Hypermedia, 1* (3) 287-299.

O'Malley, C. & Scanlon, E.O. (1989) *Computer supported collaborative learning : Problem solving and distance education.* CITE Report No. 75. Milton Keynes, UK: Open University.

Scardamalia, M. et al (1989) Computer-supported intentional learning environments. *Journal of Educational Computing , 5* 51-68

Scardamalia, M. & Bereiter, C. (1992) An architecture for collaborative knowledge building. In E. de Corte et al (Eds.) *Computer-based learning environments and problem solving* (pp.41-66). Berlin: Springer Verlag.

ERIC

Towards a New Grammar of Multimedia

MATTHEW STRATFOLD AND DIANA LAURILLARD.
Institute of Educational Technology
The Open University, Milton Keynes, MK7 6AA, UK

The grammar and use of film and television have evolved over a number of years. Now though there is possibility for a change of direction. The newly evolved power and lower cost of personal computers have made it possible for these machines to handle 'photo quality' images, video and audio. The computer enhances audio-visual media by offering ways of integrating several media and presentation formats, making use of the computer's fast processing capacity, large storage capability and random access features, to add value to the traditional media. Multimedia has been born from technology, but for it to grow and progress within education, we need a research and development effort that will explore how effective use can be made of these new possibilities.

Video in multimedia has some very different capabilities than when used in broadcast or on tape.

There is some debate as to whether interactive video warrants description as a new medium - requiring a grammar of its own, or whether it is a simple bringing together of existing media that will still keep their own unique identity and grammar. Hannafin suggests:

> "...it seems unlikely that interactive video differs from allied technology from either learning or cognitive perspectives........ (interactive video is) more of an extension of the parent technologies (computers and video) than the revolutionary innovation..."
> Hannafin (1985)

Hanson goes further than this:

> "while each medium's symbols systems and conventions of production are still relevant they may have to be adjusted in new ways to accommodate the presence in the same program of the other medium's very different symbols and conventions."
> Hanson (1990).

DeBloois takes the idea further and argues that interactive video

> "...is not merely a merging of video and computer mediums; it is an entirely new medium with characteristics quite unlike each of its composites."
> DeBloois (1982)

Now that it is technically feasible to integrate video fully in to the other media in multimedia this question needs re-examining. It is important that the use of multimedia video is investigated to see which aspects of traditional video / film grammar are still appreciated, which need changing, and what might be appropriate now.

Many of the IV developments using laserdisc technology tended to use video in the form of trigger video clips of several minutes, introducing interactive work controlled by a computer screen. This format put the learner into alternating states of reception and action. For many applications this was appropriate, and the full screen high quality video was acceptable. The quality of the video image on the early compact disc technologies, on the other hand, is unacceptable for several minutes of detailed viewing. This fact alone makes it less likely that multimedia programs will retain the style of IV. Moreover, the interest in greater learner control means that the narrative style of traditional video that has been scarcely undermined by IV will now be challenged again by CD, as learner control necessarily vitiates narrative structure. This paper considers the implications of the multimedia platform for the design of video, from the point of view of learner perception and learner control.

Video and Multimedia video - what are the differences?

If video is to be used in multimedia then what are the considerations involved in producing effective material? The considerations roughly fall into areas described in the following words: grammar, structure, interactivity, active.

The construction of video is complex - at one level production is a very intuitive process for the professional video maker, being more of an artistic than a technical process, but when examined closely the final make-up of the video is based around a common language of video which has evolved through time and is firmly linked to the physical restrictions of video's medium. To assess the way in which video in multimedia may be used differently one must address these restrictions that govern the language of video, and see in what way they have been altered in the new medium and in turn how they alter the grammar of the video and its use.

Traditionally video has always been edited for continuity, indeed 'continuity editing' is the term used in the industry to describe the way almost all video and television which we see is edited. Multimedia offers the opportunity to make a radical departure from this type of production. It would be possible to produce large sections of video which retain continuity editing in a multimedia environment, but to do so would not be taking advantage of the benefits which multimedia offers.

Many of the differences between the media are fairly obvious, others are less obvious and their effects can be less obvious still. Multimedia designers will need to develop the grammatical rules of their new medium, just as the early film and television-makers did. Eventually there may be the equivalent of the current 'bible' of the television producer 'The Grammar of Television Production' (Davis 1969). The aim of this paper is to highlight some of the issues which would need to be taken into account in such enterprise.

An investigation.

An investigation was undertaken to examine the use and role of video grammar in multimedia. The study explores some of the issues involved with transfering an original video programme to a multimedia environment, in doing so uncovers points for consideration for producers developing video destined for a multimedia platform.

The study also aims to determine the comparative values of

(i) making the structure of the material as explicit as possible;

(ii) making available orientation tools designed to assist the learner in determining their learning strategy;

(iii) adding the audio channel as a way of improving the learner's efficiency in interpreting explanations of the visuals.

The study involves the creation of two multimedia programmes using footage taken from a successful linear educational video. The first retains the structure of the existing video, giving the user only enough control to move backwards and forwards in a set of hypercard stacks with video windows, and the script of the video paraphrased in text. This version contains all the original video material in a basic multimedia format, but does not offer any complicated browsing tools, though does allow the students to work at their own pace and level.

The second also keeps the overall structure of the linear video but makes this structure more explicit, allowing the learner to locate and go directly to a section. This program also provides other orientation tools such as 'percentage complete meter', 'marking of read sections' and an 'audio summary card' option.

Care was taken to choose a video that dealt with concepts and visualisation, conveying this sort of idea being one of video's strengths.

> Television (and film, which I take to be equivalent for this discussion) is peculiarly able to convey a way of experiencing the world, because it provides a vicarious experience through dynamic sound and vision... ...For the academic who wants to convey a complex theoretical idea, television can offer a way of supplanting the process the student has to undertake in coming to an understanding of the meaning.
> Laurillard(1993)

The chosen video was an Open University introduction to science. The programme aims to foster scientific thinking in the use of models and to encourage the viewer to look at a system from a different frame of reference. The programme uses Foucalt's pendulum, the phases of the moon and retrograde planetary movement as examples. All three systems are not particularly complicated to understand, but do require that the student visualises the system in order to fully understand how it works. The subject area of the video was also chosen in order to tie into the students current science topic.

The project was carried out in a local school. The students were year eleven students preparing for their final exams (15 / 16 year olds). Pairs of students worked through the two multimedia versions. Students were also shown the original linear video version of the material. Spoken pre tests, post tests and interviews were administered and observation carried out on students who worked with multimedia versions. A written questionnaire covering the same questions was given to the students who worked with the video tape version. Ten pairs of students worked with the multimedia versions. Twenty six students watched the linear video version. Some students who had completed the multimedia versions also watched the video version.

Data from all three versions was gathered and analysed to determine learning gains and study techniques.

Points raised during the construction of windowed modular multimedia video.

Constructing multimedia video from linear video was a valuable part of the study, the process revealed a number of key points of difference between the two formats. These are listed below as considerations for a new grammar of multimedia video. The possibilities for grammar in multimedia are almost endless, an examination of what parts of grammar can be carried from the old mediums to the new must be part of a first step.

A person in a video window is too small to talk.

A talking head on video normally requires the full attention of the viewer. Existing television conventions normally require the shot length to be appropriate to what is in shot. If the subject is not talking about something that is in shot you wouldn't expect to see their full height, instead you would perhaps expect to see just the head and shoulders. If there was anything more in shot the viewer would be trying to assimilate 'why' it is there. "The ideal length of a shot is one that will just contain the essential action."Davis (1969). Similar consideration should be given to video in multimedia. Thought must be given to what else is on screen with the person in the video window, perhaps balanced against the importance of what they are saying. If what is being said is important to the program then why not use full screen, rather than a window. A video window not containing a talking person does seem to work better, and in fact may be very useful if the surrounding screen has information which is relevant, giving context or a useful contrast.

Text must match the video.

A user will normally read text on screen in preference to watching a video segment if both are provided at the same time. Observation shows that if both text and video (which is playing) are presented simultaneously the user believes, quite correctly, that the text relates to the video, and assumes that the text may be important to understanding the video and so tries to read the text first - at the cost of missing video detail. Could this indicate that the viewer believes that what is written in text is of more importance than content of the video?

Alternatives to video links between sections.

The links between sections in a video are important: it is because of these links that an argument is created and not a sequence of unrelated points. If short video sequences are to be used in multimedia, and it is not certain to which section the user will go to next, or when they will arrive, then these types of links cannot be used in the same way. It is the user who specifies their own links, since they choose when and where they are going next, based on what they have already seen. If users are to interpret the argument correctly, and make the links and decisions which the educator feels are necessary, then it may help the user to have the contents of each new node summarised as he gets it. This is done in the experimental multimedia, as the user moves from card to card, and between sections, an audible summary of the new card or section is given. In this way, even if the user flicks through the cards fairly quickly he has heard a spoken argument of his path. Thus, if the user moves between argument points, sections or cards, and the contents of the new card are made clear as soon as they arrive, then the user will be able to understand the link between the new point and the previous one while the previous information is still being considered. Users can then approach the new information with some knowledge of where they are going.

Users have greater awareness of the video's construction.

Video and especially television can be used to portray events as if they are 'real' even if they are not. Television producers make a great effort not to present anything which might draw attention to the fact that the viewer is watching an artificial construction. Of course all viewers can distinguish the difference between real life and television, but viewers are happy to allow themselves to enter the illusion of 'reality' while they watch. This is part of the attraction of television and anything that disrupts this illusion is unwelcome and can end up with users thinking about the production of what they are watching (Eg 'how did they get that shot?') rather than the contents. The users of video in multimedia are much more aware that what they see is a construction. They have control over the pace and sequence of the video. The video is split into sections and may be found in small windows rather than occupying the full screen. These factors will heighten awareness that it is a construction and will not allow the viewer to suspend disbelief. In the construction of the experimental multimedia in the present study, for example, the presenter says "we are going to *attempt* to reconstruct the experiment" the word 'attempt' has a sense of immediacy in the linear video, as the viewer is tempted to believe that the experiment might not work, and that it is being carried out in real time. This effect is lost to an extent in the multimedia as the viewer no longer feels that the experiment might fail. This may be a problem for motivational type video sequences, sequences that you are meant to associate with, or drama sequences, though this needs more investigation.

Who is the 'author'?

In the experimental multimedia program, the presenter at one point says 'we have built a model in the studio.....' this tends to jar when put in a multimedia context. There may be a number of reasons for this. It is possible that it seems wrong because the presenter in a linear

ERIC

video has some authority, being the authorial voice. On the other hand, in the multimedia version the presenter has no control over our viewing or usage of the material, and anyway he is encapsulated within or is acting as an adjunct to another author's text. It may be that for multimedia use any person speaking within a video clip must have their authority established by some associated text or audio whenever that clip appears. Further instances of this type are being investigated in the current study.

The size of a sound does not change with the size of a video window.

Sound from multimedia video which is played in a window seems out of place. During construction of the experimental multimedia it was clear that the sound accompanying the video did not function as well in a window as it did in the original video. A full screen head and shoulders shot of a person on a normal television set would make their physical image size the size you would expect them to look, taking account of perspective, if they were sitting around three feet away from you. The voice is recorded to sound natural at this distance. A similar head and shoulder shot in a video window would position you perceptually much further from the person. At this distance it would be harder for them to talk to you comfortably. The speech sounds less natural in this situation since they are at a distance but their voice sounds close. Again this needs further research.

Similarly one shot from the original video had background sound effects of bird song. The sequence showed a pendulum that was swinging oddly. It is apparent that the image and the sound do not match. This worked in the original video as a cue, or even a little joke, for the next part of the sequence which was of the same pendulum, but from a different frame of reference. The new shot revealed the reason the pendulum's peculiar motion, and for the bird song - it showed the camera man from the previous shot filming the pendulum standing on a rotating childrens' roundabout in the park, the pendulum suspended above the rotating roundabout.

To have left the bird song sound effect on the first shot in the multimedia version would have been distracting. The video in the multimedia version pauses between the first shot and the second shot where the reason for the peculiar motion is revealed. At this point the learner is asked the rhetorical question "Why does the pendulum swing strangely?". The student in answering this question is likely to consider the bird song as somehow relevant, but it is not any part of the solution. In this way what had been a background sound became too prominent.

Care must be taken when using questions to guide discovery.

It is normal practice in a video to raise topics or questions that the viewer has not yet understood. The viewer is normally quite happy with the video going on to explain what has just been said. When the progression of multimedia is under the control of the user, it is then the user's responsibility to go on to find what to do next. If users can not understand the argument fully at their present position then they will be unsure of their next step. For this reason, as in computer based learning, a default option should always be available.

Viewers of traditional video do not have this problem of course, since they have no option about direction or pacing of the video.

New options for screen format.

Many multimedia applications to date have used video windows rather than full screen video. There is an on going debate for which is more appropriate. Users of analogue video from videodisk have until recently been limited to using full screen, because to insert a video window into a computer screen really requires that the analogue signal be digitised. Users of digital desktop video had, until recently, been limited to video windows due mainly to the processing times of fetching and processing the digital data. Now though multimedia producers have the option of which to use.

ERIC

A video window in multimedia need not maintain the aspect ratio of existing television or computer monitors. Tall thin windows or circular windows are possible, in fact even irregular shapes such as a window shaped as a question-mark are possible. The question the producers must ask is 'is it appropriate?'. Davis when talking about shot angle for television suggests:

> Avoid shots that are wider than they need to be to contain the action and, even more rigourously, avoid shots that are too close to contain the action.
> Davis(1969)

There are few cameras at the moment that will record in other ratios, so any changes must be planned carefully so as to confine the action to the area to be used. It is possible to clip unwanted areas from the image during editing.

A further consideration in using video windows is what to do with the rest of the screen. If the rest of the screen is unrelated to the video, maybe just the computers' desk top GUI, or remotely related program text, then this backdrop can't really be said to add anything to the currently playing video. What the backdrop may well do is put the video in context. A graphic in the background can indicate something about the current video, and could be changed during the video, to signify changes of emphasis in the video. Keeping the current tutorial text as a background may help anchor the video's place in the course in the student's mind, telling them that this video belongs with this section.

Full frame video on the other hand takes up the whole computer screen, demanding more of the student's immediate attention. Students may expect full screen video to be more like television and feel more inclined to let it play uninterrupted. An overlaid graphical control panel may help. Perhaps talking heads in full screen video will sound more credible and authoritative. The deciding factor in making decisions like these will be the type of program and its aims. It is likely that consensus of using different formats will develop and grammars will evolve out of use.

Perhaps the most important rule in video grammar is that there must be a reason for what ever you do . Any action with out motivation will mislead the viewer. It is probably fair to say that this applies for multimedia production too.

> Any change of picture tends to attract the attention of the audience away from the subject matter and towards the technique, therefore never change it unless the next picture says something different, that has to be said, something that emphasizes a point or adds to comprehension of the audience. Never change a picture for the sake of change; it is a mere distraction.
> Davis(1969)

Again more research is still needed to examine these issues.

Working towards a new grammar will improve educational effectiveness.

Analysis of pre and post tests to determine learning gains for both versions of multimedia and the original tape based video did not reveal any difference in gain. Students were aware of the freedom which having material in a multimedia setting gave them, and some took more advantage of this than others. The problems which stopped students learning better than they did were not ones that were related to the particular media, but rather to the design of the subject material.

Students were limited by misconceptions which they did not want to part with, lack of specific goals, and shortage of motivation. Giving students the ability to control the pace of their work, or even a full set of multimedia tools and instant access to all parts of the text did not necessarily enable the students to gain any more than they could have from the tape based version. The multimedia versions in fact had not added anything more than the tape based version already had. This was part of the deliberate design aim for the programs. All the multimedia versions had added were new ways to access the material. The multimedia version was free from the restrictions of video's grammar, but had not taken advantage of the added possibilities of open to this new medium. Packaging the material in a more accessible way, or changing the medium will not necessarily improve its effectiveness. A follow-up part to this study is being worked on, a new multimedia version will have the same learning goals in the same subject area, but this time the design and structure of the material itself will be the focus. Rather than sticking with basically the old grammar and structure it will address common misconceptions, be goal oriented and offer intrinsic feedback.

Concluding Points.

Once educators decide to use video for a project for what ever reason, they will have to cover their argument from beginning to end using just video. Since all that has to be said in the video is part of the video, they are obliged to use the grammar of video throughout the production, even though at some points a continuously running linear medium might not be the most appropriate. The grammar of video is complex and there is a way of signifying all necessary actions (for example this section is drawing to a close, and a new one is about to begin). By using video in multimedia it should be possible to try to distinguish which elements of an argument are best carried or portrayed using video, and which are better taken away from video and given to other parts of the program. This type of analysis will be necessary if people are to eventually accept multimedia as a new medium with its own grammar and not simply a mixture of existing media.

References

Davis, D. (1969). *The Grammar of Television Production* London: Barrie and Rockliff.

DeBloois M.L. (1982). Principals for Designing Interactive Videodisc Instructional Materials. In *Videodisc/Microcomputer Courseware Design*, M.L. DeBloois (ed) Educational Technology Publications, Englewood Cliffs, New Jersey. 25-66

Hannafin M.J. (1985). Empirical Issues in the Study of Computer-Assited Interactive Video. *Educational Communications and Technology Journal*, 33(4).235-247.

Hanson L (1991). Producing the video for interactive video: Guide-lines for video producers and Directors. *Journal of Educational Technology systems*, 19(1) 33-40

Laurillard D (1993) *Rethinking University Teaching: A Framework for the Effective Use of Educational Technology* To be published by London: Routledge

Large Hypermedia Systems: The End of the Authoring Era

ROBERT STUBENRAUCH, FRANK KAPPE, KEITH ANDREWS
Institute for Information Processing and
Computer Supported New Media (IICM),
Graz University of Technology, A-8010 Graz, Austria

Currently, some important changes in the research community's approach to hypermedia can be observed. Some of these changes, or shifting paradigms, result from new technologies that allow the handling of ever larger quantities of information with increasing flexibility, others result from the experience that has been gained over the years with different hypermedia systems and applications.

In this paper, we focus on what we feel is one of the crucial issues in current hypermedia development: the shift from *authoring* to *information supply*. As a consequence of technical development, future hypermedia systems will be large, dynamically changing, distributed systems. The need to deal with increasing quantities of information will be the main characteristic of these hyper-mass-media systems and will be the basis by which they are judged. The traditional form of authoring an electronic document (or tutorial, presentation, etc.) involved controlling all the details of its content, layout and structure in order to best present the subject. We believe that in the future this will no longer be practicable. Only if *authors* relinquish their absolute freedom and turn into *information suppliers*, who observe some generally accepted restrictions, is the way free to deliver their information in different contexts to different users and automatically integrate it in the (possibly world-wide) information network. This heralds the end of authoring (in the traditional sense) in large hypermedia systems.

Form versus Content

Form and content are two aspects of information that are hard to keep apart in traditional media, where they usually are perceived as one unit. The *form* (the way data is presented) itself constitutes important information and is necessary to understand and interpret the *content* (the raw information or data). For hypermedia, the "form vs. content" issue is a particularly sensitive problem, because the computer can change the appearance (form) of the data (content) in almost any way we can think of. Thus the form becomes potentially dynamic: the same data may appear differently when retrieved by different users or in different contexts! Traditional authoring of electronic documents ignored this fact and tried to fit the content into a fixed form that was considered optimal for the specific subject to be presented. This notion of *fixed form* also extends to options the reader might make at reading time, in so far as these options have been explicitly determined by the author beforehand.

In this paper, we advocate a reconsideration of the role of *authoring* in large hypermedia systems in the light of the "form vs. content" issue. Through examination of current trends

ERIC

and technical developments, we discuss the implications for the processes of information supply and retrieval. We also point out a number of open problems that we think the hypermedia community should become more aware of.

From Small to Large

There is no general agreement on the features of a *true* hypermedia system. However, we claim that true hypermedia systems be potentially large (in terms of the number of information nodes they include; say more than 10,000 nodes), provide automatic maintenance of dynamically changing information, and be based on open, heavily distributed computer networks. If these criteria were applied to currently available (so-called) hypermedia systems and applications, most of them would no longer fall into the category of true hypermedia. The most popular hypermedia platforms today are stand-alone, frame-oriented, script-based systems such as HyperCard (Williams, 1987) and its relatives. Such systems hardly provide any support for automatic structuring or navigation, let alone maintenance of a dynamically changing body of information, which is crucial for handling large amounts of data. They do allow the construction of specific, carefully designed, highly interactive presentations and tutorials. However, with an ever increasing amount of rapidly changing information, it would be a Sisyphean task to manually craft such intricate detail for each piece of information.

Another aspect of the size of hypermedia systems is that orientation and navigation become mc e difficult as the size grows. These issues have been identified as crucial for the acceptance of hypermedia and have been heavily discussed in the literature (Egan et al., 1989; Nielsen, 1990). Two basic approaches stand out from the spectrum of how to access hypermedia information: *browsing* and *searching* (Marchionini & Shneiderman, 1988; Rada & Murphy, 1992). Browsing is the exploration of an information space by following associational links between related nodes. This approach may be a proper means to discover small hypermedia, but with growing size becomes increasingly adventurous. At the same time, support for efficient searching becomes essential in order to locate specific information in a large network of data, possibly to serve as a starting point for local browsing.

From Stand-Alone to Distributed, Open Systems

Most so-called hypermedia systems that run on PCs and the Macintosh (such as Toolbook, HyperCard, Guide etc.) are stand-alone systems that only allow the creation of stand-alone applications. Some other systems are based on campus-wide LANs; the most prominent of these is probably Brown University's Intermedia (Yankelovich et al., 1988; Haan et al., 1992). However, today's computer networks promise more: the fulfillment of the vision of hypermedia pioneers such as Ted Nelson, who dreamed of a global information network (Nelson, 1987).

Systems such as Hyper-G (Kappe et al., 1993a) allow to access worldwide information through Internet, integrating other distributed information systems like WorldWideWeb (Berners-Lee et al., 1992), Gopher (Alberti et al., 1992) and WAIS (Stein, 1991). Of course, this immediately raises a number of specific information supply problems. At the moment, low transmission rates limit the amount of data which can be sent, so that the only type of information typically available is text. Furthermore, due to long response times, sufficient interactivity has not yet been achieved. However, these technical restrictions will certainly be overcome in the long run. More fundamental issues are again the structure and content of such a worldwide hypermedia system and proper support for retrieval. Also, efforts must be made to avoid the "balkanisation" of the hypermedia world (Nelson, 1988).

At the moment, access to Internet information systems is organised according to the physical locations of the servers involved. One can establish a connection to a server A and then retrieve data from there. Server A may offer a connection to some other server B which in turn may offer a connection back to server A. In this way, one can at best chance upon some interesting information, but at worst can be caught in a "geographical" cycle of servers. For a hypermedia system it would make more sense to provide access according to information content, instead of the physical location of the servers. To achieve this, some generally accepted standard structure, based on keywords and categories, has to be developed. Everybody in the world connecting to a future global hypermedia system should have the same initial view of the information contained in it. Since such a system will be highly dynamic, this requirement of course raises immense technical challenges regarding efficient use of the network and maintenance of the data (Kappe et al., 1993b). Of course, issues such as ownership, privacy, usage royalties, and democracy in a global network are essential as well, but go beyond the scope of this paper.

From Isolation to Collaboration

Open distributed hypermedia systems as described above are a perfect platform to integrate with electronic mail and news services. Thus, in contrast to stand-alone systems, true hypermedia (will) provide facilities for communication and allow CSCW (computer supported collaborative work). As a consequence, the borderline between passive "readers" who only retrieve information and active "authors" who contribute information to the network, will blur. Every user will do both. This raises the question of quality and relevance control of published data, coupled with democracy considerations (who controls publications, who has access to the system and so forth). Certainly it is not desirable to have everything available that everybody has produced at any time, without any (personal) "relevance indicator".

From Single-Author to Multi-Author

Since hypermedia systems are growing, both regarding the number of nodes they contain and the distances they can bridge physically via networks, it is clear that more and more individuals are able to contribute to and access the body of hypermedia material. This fact has a consequence: as the number of contributors increases, consistency within and integrity of the data is hard if not impossible to achieve. This became obvious, for example, in the large-scale CAI project COSTOC (Huber et al., 1989), which is based on a frame-oriented system that allows the creation of arbitrary networks of frames. Although there were some written guidelines concerning the style of the lesson to be created, the 40 or so authors produced lessons of very different structure and layout according to the subject and their personal preferences. The result was that users had to adapt to many different styles of user interface. A solution to this problem might be to restrict the authors' freedom (Andrews & Kappe, 1993) by requiring them to choose some structure to be superimposed on their data. In this way, the system can automatically provide the reader with meaningful navigation support, taking advantage of its inherent organisation.

From Linking to Structuring

Links are usually considered the key feature of any hypermedia system. However, arbitrary linking has been identified as dangerous because it gives way to the production of "spaghetti networks" of information, much like the use of the goto statement in early programming (DeYoung, 1990). Various ways have been proposed to counteract this problem. For instance, there is the concept of *guided tours*, which can be considered as proposed paths through a body of information (this concept is part of the travel holiday metaphor (Hammond & Allison, 1988). However, this approach mixes content with structure and therefore does not seem generally useful for large, distributed hypermedia systems.

It is helpful to distinguish between two types of links: referential and organisational. Referential links are "real" hypermedia links in that they provide arbitrary associational connections between related nodes in different contexts. Organisational links, however, provide structural connection of nodes in a certain context. For instance, links that lead to the "next" and "previous" nodes or to a table of contents (whatever that might mean) are of that kind. (Botafogo et al., 1992) presents methods to separate referential from organisational links in an arbitrary information network by means of identifying intrinsic hierarchies and applying useful metrics. However it seems to be a makeshift solution, trying to reconstruct structure where it is lacking initially.

We believe that the use of explicit referential links across different documents or even collections (i.e. structured groups of documents; see below), can and should be mainly restricted to private annotations. For instance, one might want to establish a permanent link from an encyclopedia entry "Rome" to ones private photo of the Colosseum, in order to have quick access without the need to query for "my photo of Colosseum of summer 1992" or the like. In most other cases, explicit referential links are of limited use in large hypermedia systems. They can, in fact, be largely replaced by queries which might be presented to the user as "dynamic links", computed and temporarily used at runtime. Indeed, hypermedia systems totally absent of links (Maurer & Sherbakov, 1992) seem quite feasible.

An important feature of a hypermedia model which keeps apart content and form is that it is possible to automatically incorporate most existing data conforming to a certain structure, for example textbooks (hierarchical structure of chapters), lexicons (alphabetical order), etc.

From Authoring to Information Supply

In the CAI community, the notion of an *author* means a person who designs and develops instructional material, which is presented by a computer and which a learner or reader works through. What the author creates is is a piece of "teachware", carefully designed to meet the specific requirements of both the subject and the learner. The structure of the resulting teachware (i.e. the possible paths a student may take through the material) strongly varies with the subject to be taught and with the personal style of the author.

In the hypermedia community (which to a large extent emerged out of the CAI community), the notion of authors is still very present. When talking about authoring, most hypermedia people mean quite the same as CAI people: defining all the possible paths through an interconnected body of information (i.e. links that the reader may follow). We believe that this kind of authoring is inappropriate for *true* hypermedia systems for the following reasons. Firstly, with increasing amounts of data, it becomes simply impossible to define every link manually. Secondly, since true hypermedia are dynamic, automatic maintenance of links is required in order to ensure integrity. Thirdly, since true hypermedia are considered large, *multi-authoring* is unavoidable, which raises the problem of interface consistency across different applications. By placing restrictions on information structure, it is possible to shift control away from authors to the system itself. After all, as argued above, we do not consider referential links useful or necessary in most cases.

The role of "authors" in connection with hypermedia needs reconsideration. Perhaps the terms *information supply* and *information delivery* should be preferred to the traditional *authoring* and *presentation*. The process of information supply might be as follows: first the supplier chooses one of a selection of structures, depending on the type of data to be supplied. The system proposes a number of structures which are useful for different purposes, such as a schedules, photo albums, sound recordings, videos, phone books, poetry, etc. Then the supplier provides the actual content (mainly text, images and sounds) which has to conform to the selected structure (because it would be hard to know what to do with, say, photos put into the phone book structure). Now, when this information is delivered to a consumer, the system can take advantage of its structure and provide optimal support for navigation, orientation, searching or whatever might be useful for the specific combination of content and structure. The system presents the information in a default style that is considered useful in the context of the specific content and structure. Hence, finally the content comes to the consumer in a certain form. The default style, however, could be replaced by other styles chosen by the consumer according to personal preferences.

Of course, hypermedia systems have to provide organisation at a higher level, too. All the individual documents of different types that have been supplied must be integrated in the system. For this task we propose the notion of collections that may contain single documents as well as other collections. The set of collections builds an inclusion hierarchy which can be searched for node names, keywords, attributes etc. In (Maurer & Sherbakov, 1992) a data model is evolved that extends the concept of collections to a general model that includes the idea of structure discussed above.

Clearly, this shift away from authoring is in fact restrictive. Information suppliers have less control over how the reader will finally perceive their data. However, this can also be considered an opportunity to concentrate on pure content and a liberation from the burden of being responsible for all aspects of preparing electronic documents. After all, it is hard to find individuals with the joint talents of being experts in a specific field as well as in layout, electronic publishing, interface design, etc. Information consumers, in turn, might appreciate the new freedom to discover docuverses in their own preferred way without being patronised by authors (although many people might prefer being patronised to having the responsibilities that come with freedom). In the context of CAI, (Hartog, 1989) identified such a shift of paradigms from *process control* (program controlled, possibly "intelligent" tutorials that guide the learner) to *information resources* to be explored by learners.

Nevertheless, outstanding authors in the traditional sense, or, more likely, experienced electronic publishing teams, will still make contributions that do not necessarily conform to all the structural restrictions discussed above. This is highly desirable and must be supported by hypermedia systems. However, it is important to see that in very large, distributed systems, only a small minority of users will be able to contribute outstanding material (in terms of contents, structure, interface, etc.) worthy of being apart from proposed and accepted standards. After all, production of high-quality electronic documents is very cost intensive, involving, among other things, domain-dependent human experts. The large majority of average users will appreciate well-considered standard structures which they can

ERIC

take advantage of. Once such well-considered standard structures exist, only very few suppliers will be able to afford the cost and the risk of following their own differing approach; the challenge is to develop these standards.

From Presentation to Multi-Metaphor Selection

In a large, distributed hypermedia system, different viewer software running on different platforms will provide different means of presentation. Following on from the above, the consumer of a hypermedia system potentially has the option of *customising* the retrieved data according to his or her personal needs. That means the user can choose between different presentation *metaphors* (Davies et al., 1991; Erickson, 1990). However, this is a critical issue because it is affected by the "form vs. content" issue. Selecting a metaphor for presentation means changing the form of the presented information. Thus it is a problematic freedom to let users decide on form (and, as stated above, many users might just prefer being patronised). Furthermore, the usefulness of interface metaphors in general has been questioned, because they hinder new developments by sticking to familiar "real-world" analogies such as desktops, books, travels etc. Therefore, in (Nelson, 1990), interface metaphorics are put on the list of mistakes and elements of bad design.

Conclusion

We showed that for very large hypermedia systems (in terms of number of nodes, users and degree of distribution through networks) it seems indispensable to introduce a certain amount of discipline regarding the structure of information. We advocate the sensitive separation of content from structure and the renunciation of arbitrary linking as far as possible. The basic problem of this approach is that there is no commonly agreed-upon set of structures that are considered appropriate for all kinds of electronic documents.

Hypermedia is still in its infancy and not all of its concepts have been worked out. If hypermedia systems of the type described in this paper (large, distributed, dynamic, linkless and structured) are going to be successful in the sense that they satisfy the needs of a large user community (not necessarily the same as the needs of hypermedia developers!), information suppliers as well as consumers must be aware of and observe the characteristics of these new media. A new kind of literacy plus appropriate rhetorics for hypermedia must evolve and become commonplace. Though this will be a long-term process, the traditional notion of authoring will be superceded by that of information supply.

References

Alberti, B., Anklesaria, F., Lindner, P., McCahill, M., & Torrey, D. (1992). The internet gopher protocol: a distributed document search and retrieval protocol. Available by anonymous ftp from boombox.micro.umn.edu in directory pub/gopher/gopher_protocol.

Andrews, K., & Kappe, F. (1993). Strait-jacketing authors: user interface consistency in large-scale hypermedia systems. In Frei, H. P., & Schäuble, P., editors, *Hypermedia '93, Zurich, Switzerland* (pp. 130–137), Springer, Berlin.

Berners-Lee, T., Cailliau, R., Groff, J., & Pollermann, B. (1992). WorldWideWeb: the information universe. *Electronic Networking: Research, Applications and Policy,* 1(2), 52–58.

Botafogo, R. A., Rivlin, E., & Shneiderman, B. (1992). Structural analysis of hypertexts: identifying hierarchies and useful metrics. *ACM Transactions on Information Systems*, 10(2), 142–180.

Davies, G., Maurer, H., & Preece, J. (1991). Presentation metaphors for very large hypermedia systems. *Journal of Microcomputer Applications*, 14(2), 105–116.

DeYoung, L. (1990). Linking considered harmful. In Rizk, A., Streitz, N., & André, J., editors, *Hypertext: Concepts, Systems and Applications; Proc. ECHT'90* (pp. 238–249), Cambridge University Press.

Egan, D. E., Remde, J. R., & Landauer, T. K. (1989). Lost in hyperspace: cognitive mapping and navigation in a hypertext environment. In McAleese, R., editor, *Hypertext: Theory into Practice*, (pp. 105–125), Blackwell Scientific Publications Ltd.

Erickson, T. D. (1990). Working with interface metaphors. In Laurel, B., editor, *The Art of Human-Computer Interface Design*, (pp. 65–73), Addison-Wesley, Reading, Massachusetts.

Haan, B. J., Kahn, P., Riley, V. A., Coombs, J. H., & Meyrowitz, N. K. (1992). IRIS hypermedia services. *Communications of the ACM*, 35(1), 36–51.

Hammond, N., & Allison, L. (1988). Travels around a learning support environment: rambling, orienteering or touring? In Frye, D., & Sheppard, S. B., editors, *Proc. CHI'88* (pp. 269–273), ACM.

Hartog, R. J. M. (1989). Computer assisted learning: from process control paradigm to information resource paradigm. *Journal of Microcomputer Applications*, 12(1), 15–31.

Huber, F., Makedon, F., & Maurer, H. (1989). Hyper-COSTOC: a comprehensive computer-based teaching support system. *Journal of Microcomputer Applications*, 12(4), 293–317.

Kappe, F., Maurer, H., & Sherbakov, N. (1993a). Hyper-G – a universal hypermedia system. *To appear in: Journal of Educational Multimedia and Hypermedia 2(1).*

Kappe, F., Pani, G., & Schnabel, F. (1993b). The architecture of a massively distributed hypermedia system. *Electronic Networking: Research, Applications and Policy.* Submitted for publication.

Marchionini, G., & Shneiderman, B. (1988). Finding facts vs. browsing knowledge. *IEEE Computer*, 21(1), 70–80.

Maurer, H., & Sherbakov, N. (1992). Hypermedia systems without links. *Hypermedia.* Submitted for publication.

Nelson, T. H. (1987). *Literary Machines, Edition 87.1.* The Distributors, South Bend, IN.

Nelson, T. H. (1988). Unifying tomorrow's hypermedia. In *Online 88 Information: Proc. 12^{th} International Online Information Meeting* (pp. 1–7).

Nelson, T. H. (1990). The right way to think about software design. In Laurel, B., editor, *The Art of Human-Computer Interface Design*, (pp. 235–243), Addison-Wesley, Reading, Massachusetts.

Nielsen, J. (1990). The art of navigating through hypertext. *Communications of the ACM*, 33(3), 296–310.

Rada, R., & Murphy, C. (1992). Searching versus browsing in hypertext. *Hypermedia*, 4(1), 1–30.

Stein, R. M. (1991). Browsing through terabytes – wide-area information servers open a new frontier in personal and corporate information services. *Byte*, 16(5), 157–164.

Williams, G. (1987). HyperCard. *Byte,* 12(14), 109–117.

Yankelovich, N., Haan, B. J., Meyrowitz, N. K., & Drucker, S. M. (1988). Intermedia: the concept and the construction of a seamless information environment. *IEEE Computer,* 21(1), 81–96.

ERIC Full Text Provided by ERIC

ISLE: A collaborative project to build an Intensely Supportive Learning Environment

Peter Sykes[1], Joachim Schaper[2], Terry Mayes[1], Hilary Palmen[1], Vic Marcipont

[1] Institute for Computer Based Learning, Heriot-Watt University, Riccarton, Edinburgh, EH14 4AS, United Kingdom Phone [+44] (31) 449 5111,

[2]: Digital Equipment GmbH CEC Karlsruhe Vincenz-Prießnitz-Str. 1, W-7500 Karlsruhe, Germany Phone [+49] (721) 6902-34

BACKGROUND

The ISLE project started in January 1992 as a collaborative venture between Digital and Heriot-Watt University. The concept described in the name of the project is that of an 'Intensely Supportive Learning Environment'. The project was founded on the common recognition that there still exists a wide need for an environment that will integrate a whole set of key learning technologies in adult learning and higher education. The initial ingredients for the creation of this environment are:

- **Athena:** The Athena project [GC 91] at MIT ran for 8 years. Funded by Digital and IBM, its remit was to implement a computing environment for educational use. The ISLE project has taken some of the ideas and outputs from the Athena project as a base from which to further develop the concept of a supportive learning environment. Digital has released DecAthena, which is the system management and communications aspect of the MIT Athena project. Heriot-Watt University has adopted DecAthena to manage its rapidly growing campus network of workstations.
- **Nestor:** The Nestor project has been underway at Digital's Campus Engineering Centre at Karlsruhe, Germany since 1988. Nestor [MS92] provides a suite of tools to generate courseware, present it to students and allow them to share a computing environment to discuss the material. Nestor provides the project with much of the software on which to base the ISLE. The emphasis of Nestor is on a network based collaborative multimedia environment and its focus on instructional design support and courseware construction complements the services of Athena and contributes to an initial ISLE prototype.
- **Heriot-Watt University:** whose institutional policy is committed to advanced teaching developments using technology to provide higher quality and more efficient education. The Institute for Computer-Based learning at Heriot-Watt has been created in support of this policy
- **User-centered iterative design philosophy:** ISLE is based on the view that empirical testing, rather than the availability of new technology, must guide the design. However, even with the considerable resources at our disposal empirical iterations are limited by many practical constraints. ISLE's user population will include the traditional University roles associated with both teachers and students, with teachers engaging in the creation of small-scale courseware and students producing a variety of coursework. Systematic user surveys of these activities are central to ISLE development.
- **A constructivist view of learning:** ISLE is conceived essentially as an environment in which learners will enjoy maximum support for their activities [KOM92]. This implies a very different view of the role of technology than that associated with the notion of the computer as a multimedia device for the delivery of courseware.

A Context of Rapid Change

The overall context for ISLE is one of rapid change. The most pressing aspect of this in UK higher education is the need to maintain standards in the face of rapidly expanding student numbers accompanied by a fall in unit resource. The recent report of a working party set up by the Scottish University Principals [SCFC92] has emphasised that technology-based methods need to be seen as one aspect of a general shift towards open learning. An interesting side effect of the demands of this change is the new found willingness of Universities to abandon the long-standing "Not Invented Here" attitude and to co-operate in the development of both methods and materials. A recent UK initiative (The Technology in Teaching and Learning Programme - TLTP) has awarded contracts to consortia varying in size from 2 to 40 institutions to develop courseware in almost every field of Higher Education.

The ISLE project is also being undertaken in a time of rapid development in University / Industry cooperation in research. Digital has formed its partnership with Heriot-Watt through a conviction that partnerships with users are more likely to meet their eventual needs. The goal is to better design the software by developing and testing it in the context in which it is intended to be used. Therefore the evaluation strand in ISLE is very highly emphasized and the concept of user centered design is used throughout the development of the system.

ISLE

If we performed an observational study of what it is that students and their lecturers actually did in their working activities, then we would presumably focus more on the student's study and the tutor's office than on the lecture room or the lab. We would do so because most of the business of both teaching and learning takes place primarily in these highly personalised, often paper-based environments. It is here that the student will rewrite lecture notes, draft essays, take notes from books and papers, file handouts, write up lab reports, carry out problem-solving assignments, collect previous exam papers, and as the exam approaches may experiment with techniques - such as structuring notes in new ways - that directly aids the understanding (or simply the memorising) process. The lecturer will also perform a surprisingly similar set of activities - taking notes from books and papers, drafting new lectures, updating old ones, compiling lists of references, preparing new handouts, and producing assignment and exam questions. For both of them we would observe a highly developed personal system of storing and retrieving information. Each of us creates a world for learning structured by a kind of dynamic filing system that depends entirely on our constantly developing understanding of what all the 'objects' (chunks of info / knowledge) mean and how they relate to each other. What lies at the heart of our conception of an ISLE is a system that can allow just such a highly personalised and highly flexible way of storing the objects or 'chunks' of learning. What characterises these objects are their variety. Much of the material will be in the form of notes - personal notes from lectures and private reading, but also information provided explicitly for learning from the tutor, in the form of 'handouts'. On top of these the learner builds elaborations, as understanding deepens, and some of these elaborations will be copied (pasted) from books and articles, together with personal annotations. As the student (or the tutor) constructs a larger and larger edifice of bits and pieces of knowledge then patterns of links begin to emerge, and the material begins to take on structure. Frequently one structure won't be sufficient. Sometimes the learner will want to undo all the structure so far, and try a new one. ISLE will need to support all this activity with an integrated set of tools. We have used the term LearningBase to refer to the main application.

We have used the following classification to place the contents of the LearningBase into 3 main categories:

- **Primary Courseware:** This is professionally produced and distributed by the publishing industry. It is highly organised multimedia courseware, simulations, video material etc.
- **Secondary Courseware** This is less structured material, worked on and tailored by the individual tutor or student. It is, for example, the copy of a paper from a journal, the tutor's own lecture notes, or local annotations to primary courseware. Most importantly it is the students own created material as well as the material supplied for the course.

- **Tertiary Courseware** This is the material generated by discussions about the course and the sharing of the learning process by the students. It is a record of the questions to tutors and the answers, a record of the threads of argument developing in a computer mediated conference, captured to become a resource for subsequent students.

Implementation

To implement this concept we have developed an initial ISLE architecture illustrated in Figure 1. There are a number of media presenters and editor suites with the task of handling the users' interactions with objects of a particular media type. The selection and order of the objects these presenters display to the user is determined by one of a number of structuring tools. At the lowest level we have an object store which maintains details of the attributes of these media objects, with a librarian function to locate them. The purpose of creating a distinction between these tools is to separate the function of structure and display as much as possible so that objects created as a part of one structure may be re-used as part of another.

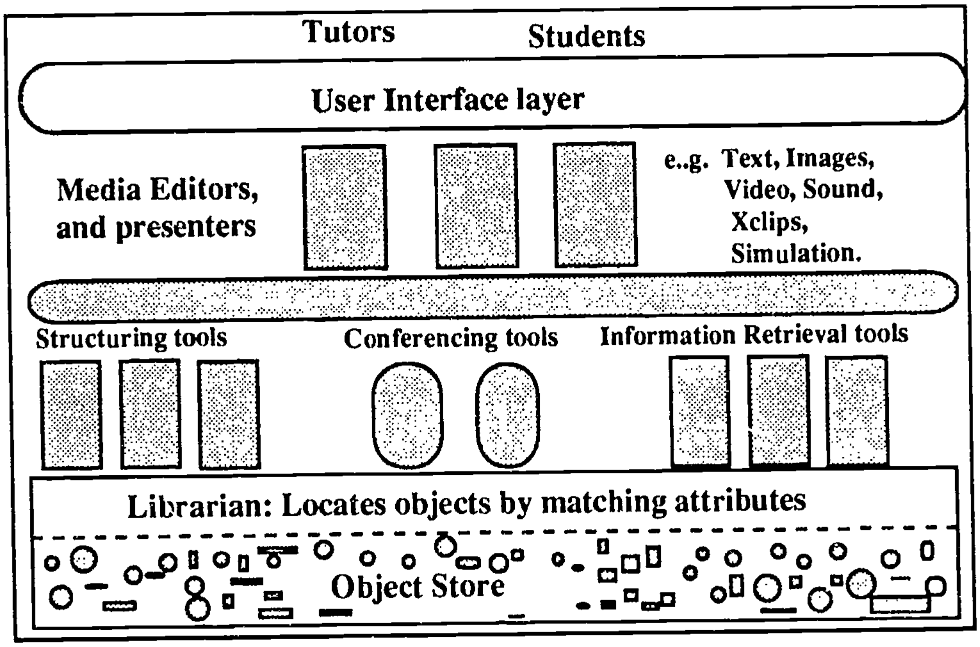


Figure 1: Isle Achitecture Overview

The media tools give form to the objects in the ISLE environment. The media types will be as rich as each individual ISLE can support. The media tools may be specified according to the requirements of the media to be presented and the preferences of the user. The media types we envisage we will support are Text, Graphics, Simulations, XClips [Nes91] Audio and Video. [XM92].

The structuring tools are the key aspect of ISLE. They give context to the objects that are to be presented to the student. A number of structuring mechanisms have been selected and are undergoing testing.

- A structured course design environment and course interpreter as provided by Nestor.
- A relevance matching and ordering information retrieval system modeled on StrathTutor [MAY88] [KIB89] and the ordering system in WAIS [KAH89] by which the user retrieves an object in the object store. There are many possibilities here. A vital issue is one of design-

ing an indexing mechanism in such a way as to give the user maximum flexibility for subsequent retrieval without an unacceptable overhead when an object is created and stored.

- A 'cognitive tools' approach where a student is able to process information in a way that promotes deeper understanding. [KOM92]. This may involve some form of concept mapping, or other special-purpose knowledge acquisition tools, but it is worth noting that traditional authoring tools can also be effective as cognitive tools when placed in the hands of learners themselves.
- A notebook metaphor supporting the "acquisitive" mode of learning. This is also a placement tool placing objects in the LearningBase under the title of the notebook and assigning attributes to them.
- An on-line conference structure of topics and threads of discussions within those topics.

Communications

ISLE includes the model of context sensitive communications used in the AnswerGarden [ACK90] where queries about the LearningBase are automatically directed by email to a nominated expert in the particular area being queried. ISLE extends this by providing a facility to ask a question of the tutor, the rest of the class or the references supplied as a part of the course assets. This is supported by the provision of one-to-one, one-to-many, synchronous and asynchronous communications. An early prototype of ISLE has supported this with electronic mail to the tutor, and Internet Relay Chat (IRC) [JO91] a real time text based multi user conference to the rest of the class. A usenet news group and a mail list were provided for off-line communications to the class, the former used primarily for bug reports. Finally, an archive of a computer mediated conference concerning the previous year's course was provided as a local WAIS service. In order to allow groups to communicate synchronously using standard X-Windows applications a sharing component 'shX' [MA90] will be used which allows a learner to share the state of the window and the interactions therein with any other learner or tutor. This will be complemented by audio/video conferencing tools. Telepresence,[MAT92] is potentially useful to make ISLE users aware of who is using the course materials and who is willing to discuss them or collaborate over their use.

Object Store and Librarian

The object store manages the objects in ISLE for both the tutors and the students. The former typically creating publicly available objects, the latter working on their private objects. It supports the tagging of these objects with attributes which may be public or personal to the individual attaching that attribute to that object. Objects in the store include the individual media entities such as text, images, pointers to external information sources and pointers to the communications paths, such as a pointer to the IRC channel, an audio conference or a Usenet conference. Objects may be composites such as collections of linked objects or a thread in a Usenet conference.

The librarian manages the information retrieval aspect of the LearningBase. It must be able to retrieve objects by their attributes, where the attributes may be generated by the acquisition tool or structuring tool that put the objects in the LearningBase or be generated dynamically using whichever method is appropriate to the media type.

Development

An aim of the ISLE project is to develop this learning environment with minimal software development by providing an infrastructure in which to envelop the tools that will be required, and which are being developed elsewhere. The goal is one of interoperability between different hardware and software platforms so it is important to adhere to standards where possible. As noted earlier the project's approach is an iterative development approach based on a developing understanding of the tasks associated with the learning environment.

The ISLE development process will supplement the traditional design, develop, deliver model. Initially, the definition of the design requirements is through input from observation of

ERIC

field studies, through user surveys, through studying the use of existing advanced learning technologies, and through surveying other available technologies for inclusion. These requirements will evolve as both the available technology, and the development teams' understanding of the task evolves. Initial prototypes are being evaluated by pilot testing.

TECHNOLOGY

The technology proposed for ISLE is derived from several sources: Athena, Nestor, a set of information management tools and the rapidly growing number of network communications tools. In the following we will outline the components with short descriptions. In the case of Nestor we will go into more detail to emphasis the role of the authoring tools as well as the presentation tools.

Nestor

Nestor is a set of tools which are designed to cater for instructional design, multimedia presentation and collaborative work. The Nestor concept is documented in the general Nestor reports [NES 90, 91] and in reports which focus on hypermedia [MM 91], databases [DL 91], multimedia [GB 91] and collaboration [TR 90] aspects. The current description is of the tools that are available now and are derived from extensions to current Digital products or prototypes that are based on public domain software. This list of tools is shown in Table 1:

The overall structure of how the Nestor tools are used during the design, development and delivery of courseware is shown in Figure 2

Table 1: Nestor components

Tool	Function
Instructional Design Editor	To perform and document an instructional analysis
Course Structure Editor	Generating a flow oriented course structure
Course Interpreter	Running an instance of a course
Media Presenter:	
Bookreader	On-line reading of text, graphics and images
Software Motion Pictures	Video done in software
Audio	lofi audio
Xclips	Samples of recorded X applications
Learner Interaction Elements	User interface and test elements to traverse the course
LinkWorks	Generic inter-application hypermedia link service
shX	Allows real time sharing of any X- application

Design

During the design stage the Instructional Design Editor (IDE) [Nes 92] is used to document the instructional analysis (also known as the domain analysis) as a network of nodes and links. The IDE also provides a set of high level instructional strategies which are currently "tutorial" implemented as progressive deepening, "Guided Tours" and "Quiz". These strategies create a specific default flow for the Course Structure Editor.

Courseware Development

During the development phase, the media content is provided using media editors such as DECWrite to edit the electronic books for Bookreader (see delivery), image, audio and video editors. All the media contents are assembled for the course flow using the Course Structure Editor (CSE) which receives a generic course structure from the IDE depending on the selected instructional strategy.

The course flow is controlled by the student by the interactions described in the User Interface Builder (UIB) module. As this component is still under development, there is a limited set of pre-defined building blocks to create new instructional transactions such as multiple choice questions or fill-in-the-blank forms. The CSE generates the input for the Course Interpreter which at run time keeps the representation for the course state and a rudimentary learner model. The major advantages of the partitioning of the delivery system in various delivery servers is that the system can be configured according to the characteristics of a certain system . It also allows us to add new servers such as a simulation server for driving complex engineering simulations.

Courseware Delivery

The Course Interpreter (CI) implements the engine that drives the overall course presentation. It triggers the mediaserver to open up Bookreader pages, present colour images, and various multimedia clips such as sound, video or Xclips. The CI also triggers the user interface server to present all graphical user interface elements, the learner needs to navigate throughout the course and the related material. It also provides the instructional transactions elements such as tests (e.g. multiple choice or fill-in-the-blank questions). The CI also generates logs from a certain learner session, which can be later used to be evaluated. This generated feedback can then be used to enhance the authored courseware using the IDE/CSE.

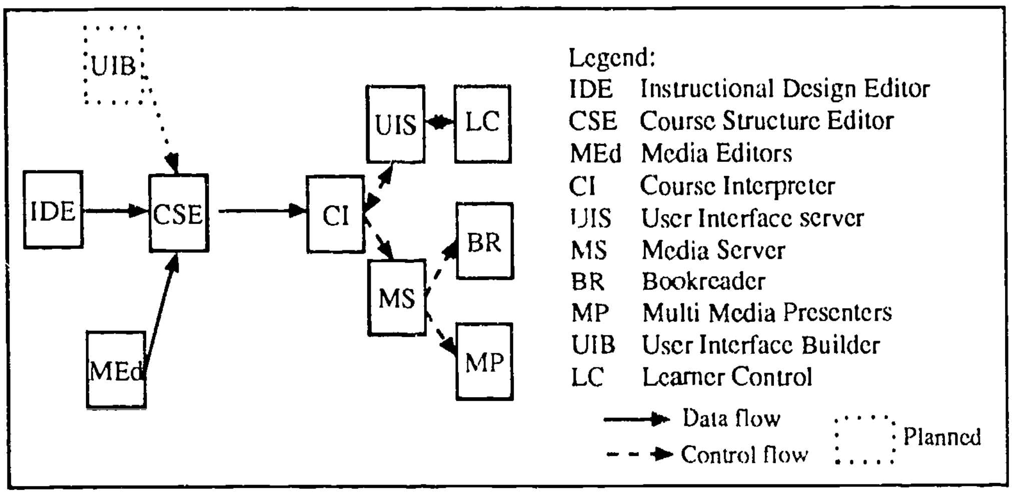


Figure 2: Nestor tools

Notebooks

The students interface to the object store will be provided by a Notebook metaphor in which objects may be generated using the appropriate media editors and placed on the pages of the notebook. Alternatively, objects may be located in the publicly available collection in the object store and transferred to the pages of the student's individual notebook. Notebooks may be produced within ISLE for the individual, the course, the class, or the whole faculty. We envisage features such as overview page, navigation aids, diary pages, versioning and automatic updating. However, it is an essential part of the ISLE philosophy of development that design and implementation of such features is derived from the needs of the users, not from the imagination of the developers.

Information retrieval

ISLE will support a model of information retrieval based on attribute matching and relevance feedback. Each of the objects will have a number of attributes. These may be the attributes supplied by the creator of the object as well as those supplied by the present user of the object. Attributes may also be dynamically generated by searching the object - if appropriate to the class of object and/or its media type. Let us consider a student who has discovered something of interest in a Nestor course and wants to know more. By locating objects in the store with attributes similar to those of the current object ISLE should provide more information. This search may then be refined by selection of either (a) the specific attributes of interest or (b) a relevant object in the retrieval list whose attributes may be used to refine the search.

Summary

This paper has provided a brief overview of the joint project between Heriot-Watt and Digital which is focused on the elaboration of existing technology to build a system that supports an intensely supportive learning environment for higher education. It is sufficiently generic to apply in other areas of education and in training. The system is intended to be built incrementally and to allow a continuous refinement for both the system itself as well as the learning infrastructure that is provided by ISLE. Many questions have not been addressed in this short paper, and many remain to be resolved in the project itself. Not least of these is the extent to which a generic ISLE is a valid concept. Learning occurs as a by-product of motivated activity and the wide range of these activities themselves need to be supported by any effective learning environment. Can a generic environment adequately support the great variety of such activities? It can be argued that would it be more profitable to design an ISLE for, say, mathematics, and one for botany, and one for French, and then to derive a generic system from the similarities. Such an approach may indeed emerge from our empirical work and it remains to be seen how far the goal of a generic solution can be sustained.

Acknowledgments

The ISLE Project is supported by the Distance and Open Learning programme of Digital's European Technology Management Group. Particular thanks are due to Professor Alistair MacFarlane, Principal of Heriot-Watt University, who was the originator of the ISLE concept, and to the following of our colleagues who have been associated with the project: Dave Beaumont, Ken Currie, Tony Eldridge, Stuart Fleming, Ben Frommherz, Alistair Kilgour, Richard Kirby, Ray McAleese, Patrick McAndrew, Chris Owens, Stephen Peters, Roger Rist, Carmel Smith, Florin Spanachi and Jim Waters.

References

[ACK90] Ackerman, M., Malone, T.W., *Answer Garden: A Tool for Growing Organizational Memory*. Proc: Conference on Office Information Systems. ACM Press, Cambridge 1990

[DL 91] Dürr, M., Lang, S.:*A Data Schema Approach to Multimedia Database System Architecture*, Proc. Intl. Conf. on Multimedia Information Systems, Singapore, Jan. 1991

[SCFC92] The Committee of Scottish University Principals *Teaching and Learning in an Expanding Higher Education System*. SCFC: Edinburgh 1992

[GB 91] Blakowski, G.:*The MODE-FLOW-GRAPH: A processing Model for Objects of Distributed Multimedia Applications*, Proc. Int. Sym. on Communications, Tainan, Taiwan, Dec. 1991

[GC 91] Champine, G.A.:*MIT Project Athena A Model for Distributed Campus Computing*, Digital Press 1991

[JO91] Oikarinen, J., *Internet Relay Chat Manual.* University of Oulu, Finland

[KAH89] Kahle, B., *Wide Area Information Servers Concepts.* Thinking Machines Technical Report TMC-202

[KIB89] Kibby, M. R., Mayes, J. T. (1989) *Towards intelligent hypertext.* In R. McAleese (ed.) Hypertext: theory into practice. Norwood, New Jersey: Ablex.

[KOM92] Kommers, P. A. M., Jonassen, D. H., Mayes, J. T. *Cognitive Tools for Learning.* Springer-Verlag: Berlin Heidelberg 1992

[MA90] Altenhofen, M., Neidecker-Lutz, B., Tallett, P.: *Upgrading a Window System for Tutoring Functions*, Proc. EX (The European X Window System Conference and Exhibition), Nov. 1990

[MAT92] Matsuura, N., Fujino, G., Okada, K., Matsushita, Y. : *Supporting Encounters and Interactions in a Virtual Environment*, Proc Inet 92 (The International Networking Conference) , June 1992

[MAY88] Mayes, J. T., Kibby, M. R., Watson, H. *StrathTutor: the development and evaluation of a learning-by-browsing system on the Macintosh.* Computers and Education, 12, 221-229. 1988

[MM 91] Mühlhäuser M.: *Hypermedia-Navigation as a Central Concept for Instructional Tool Environments*, Proc. CALISCE'91 Lausanne, Swiss

[MS92] Mühlhäuser, M., Schaper, J.: *Project NESTOR: New Approaches to Cooperative Multimedia Authoring / Learning.* Proc. ICCAL'92, New York: Springer 1992.

[NES90] Coyle, K., Spanachi F.: *Guided Tour through The Nestor System*, CEC Karlsruhe, Technical Report 1990

[NES91] Nestor team: *Nestor System - Course Authoring and Learning- A New Approach*, CEC Karlsruhe, Technical Report 1991

[NES 92] Haida, A., Schaper J.: *Instructional Design Editor - User's Manual*, CEC Karlsruhe, Technical Documentation 1992

[TR 90] Rüdebusch, T.:*Requirements and Concepts for a Group Interaction Environment*, HIS'90 - Sixth Sym. on Human Interface, Tokyo, Oct. 1990

[XM 92] XMedia Tools: *Video Toolkit Developer Guide.* Digital Equipment Corp. 1992

Optimal Presentation of Links in Large Hypermedia Systems

Ivan Tomek[1], Hermann Maurer[2], Mostafa Nassar[1]
[1]Jodrey School of Computer Science
Acadia University, Wolfville
Nova Scotia, Canada, B0P 1X0
fax: (902) 542 2201,
e-mail: ivan.tomek@acadiau.ca, mostafa.nassar@acadiau.ca
[2]Institutes for Information Processing
Graz University of Technology
Schieszstattgasse 4a
Graz, A-8010, Austria
e-mail: hmaurer@iicm.tu-graz.ac.at

Large hypermedia systems such as computer-based learning environments may contain an overwhelming number of links. To make such systems useful, links should be presented to the user in an order reflecting their value for the viewer. We propose an approach to the ranking of links based on the assumption that the user's goal is to select the link that maximizes his current knowledge of the hypermedia network. We describe several quantitative measures that could be used for such ranking, comment on their feasibility, and discuss some modifications and extensions.

Introduction

While the notion of links connecting related nodes is the central concept of hypertext, it presents several problems. One of them is the unstructured topology of the information space that it creates, a problem which has been related to that of GoTo statements in computer programming. The problem of uncontrolled linking has been addressed by many authors and an approach similar to the remedy for "spaghetti code" consisting of the introduction of higher level aggregates grouping related nodes and links has been proposed. A recent paper (Botafogo91) proposes graph-theoretical methods that could simplify the task of creating such aggregates by automatically identifying node-link clusters.

While formation of hierarchical structures makes information stored in large hypermedia systems easier to understand and use, the number of links emanating from individual nodes may still be overwhelming. The question then is how to present these links to the user so that he can make good use of them. This problem has not received adequate attention in the existing literature.

BEST COPY AVAILABLE

ERIC

To put the problem into a perspective, consider a driver reaching a crossing from which a large number of continuations is possible. If the driver's "goal" is to wander around randomly, any choice is as good as any other. If, however, the driver wants to reach a particular location, then the available alternatives can be ranked, for example, on the basis of the amount of time that it will take to reach the desired destination.

The traveler in a large hypermedia system also may have to make choices from a very large number of available links. If he is only browsing the system without any particular goal in mind, there is probably not much that the system can do for him. If, however, the hyperm dia traveler has a clearly defined goal, then systematic ranking of available choices is possible. We concentrate on this second case. In particular, **we assume that the goal of the user is to select the next node so as to maximize his current knowledge.**

It should be noted that this formulation covers some of the situations that might appear to fall into the random browsing category - the point often is to properly formulate the kind of knowledge that the user is seeking: As an example, while a person seeking information about differential geometry obviously falls into the category of users with a goal, so does, for example, a person looking for otherwise unspecified "unusual" information present in the system. Both are hoping to maximize their gain of knowledge and the difference between them is how they evaluate the information stored in individual nodes. In this light, the case that we are investigating covers a wide range of situations.

If we accept the premise that available links should be presented in such an order that their traversal increases the current knowledge of the user, the question is how to evaluate this cost functional. The prerequisite clearly is that we must be able to express the value of information stored in a hypermedia system numerically. We described our first thoughts on how this could be done in (Tomek92) and the following sections expand our original proposal.

Evaluating the information present in a hypermedia space

If we ignore aggregates forming higher level hierarchical structures, a hypermedia space consists of nodes and links. These nodes are typically textual but they may also have the form of graphics, digitized sound, digitized video, and so on. In this paper, we are restricting our attention to those cases where it is reasonable to say that the value of nodes and possibly links as well can be expressed quantitatively. In this sense, the hypermedia space is related to a knowledge network consisting of nodes and their relations, and we can talk about the value of information present in it.

Up to now, the question of how to measure the value of information quantitatively does not seem to have been addressed and we propose to approach it intuitively. Clearly, one of the factors that determine the value of information present in a knowledge network is the value of the facts. However, the links between the facts are also valuable. Consequently, it seems intuitively acceptable to state that the total value of knowledge present in a hypermedia space is a weighted sum of the values of facts and the values of links:

$$K = Sum(f_i) + c.Sum(l_{ij})$$

where f_i is the value of fact i, l_{ij} is the value of a link between facts i and j, and c is a weighting coefficient expressing the relative weight that we attach to facts in relation to the weight of links. The sum is over all nodes and links in the system.

Let K_0 be the value of the current knowledge of a user with respect to the current hypermedia space, essentially the value of information in the system already perused by the user. K_0 is obtained by evaluating the above formula over S_0, which is that part of the complete space known to the user at present. Let K_x be the value of the space obtained by adding to S_0 node i reached by traversing link x. To obtain the desired ranking of currently available links, we should calculate the values of K_x over all links x emanating from the current node, and the link that gives the largest value of K_x should be presented as the top choice among available links because it adds the most to the user's current knowledge.

What we have said so far seems intuitively acceptable but it is not clear how to calculate the value of facts and links and how to select the value of coefficient c. At present, we don't have complete answers to these questions but we propose some possibilities in the following sections.

Values of facts and links

We propose that the value of a fact consists of its **intrinsic** value (the value of the fact itself) and the sum of the weighted values of all facts linked to it. In other words, we claim that a fact is more important if it is known to be related to other facts. We call the value of a fact that captures this context its **derived** value and calculate the derived value f_i of fact i as

$$f_i = F_i + k_f.Sum(v_j)$$

Here F_i is the intrinsic value of fact i, k_f is a scaling coefficient that determines the relative

ERIC

importance of related facts with respect to the intrinsic value of the fact itself, and the sum is over all facts j directly reachable from fact i via a link.

At present, we don't know how to determine the intrinsic value of a fact and we doubt that there is an algorithmic way to do it. This is partially because the intrinsic value of a fact depends on the context in which the user is using the system. As an example, the same mathematical fact may be of great value to a student of topology but of little value to a student of algebra or to the curiosity seeker mentioned above.

While we do not have a definitive method for the calculation of intrinsic values, some possibilities for obtaining them are as follows:

- Assign all facts the same intrinsic value. When this approach is used to calculate derived values, we obtain greater derived values for facts that are more densely connected, in other words, playing a more central role. This results in ranking nodes on the basis of their connectivity.
- Allow the author of a fact added to the system to assign intrinsic value by subjectively weighing the new fact against other facts in the hypermedia space.
- Allow users to influence the derived value, for example by building a voting mechanism into the hypermedia system.
- Assign intrinsic value on the basis of the degree of certainty about the validity of the fact. Note that this might result in negative intrinsic values for facts whose validity is suspicious.

As for **values of links** between nodes, one can again decompose the value of a link into its intrinsic part and a component derived from the context. It seems natural that the intrinsic value should be the same for all links except in situations where links are typed. In this case, the intrinsic value of a link might depend on its type and on the purpose for which the user is using the system. Further, we propose that a link relating two important facts is more valuable than a link relating two unimportant facts. On this basis, the derived value l_{ij} of link ij connecting facts i and j can be expressed as

$$l_{ij} = L_{ij} + k.(f_i + f_j)$$

where L_{ij} is the intrinsic value of the link.

Our final comment concerns values of **scaling coefficients**. Although we cannot offer much quantitative advice yet, values smaller than 0 would make facts with known relations to

ERIC

other facts less valuable than facts without known relations and this does not seem logical - unless the relations themselves are dubious. Consequently, the scaling coefficient should not be negative. Further, since the calculation of derived values results in including first order, second order, and more distant neighbors through dependencies via the resulting set of linear equations, values greater than 1 effectively make more distant related facts more important than immediately adjacent facts which again does not make sense. Altogether, scaling coefficients should be between 0 and 1.

Implementation and feasibility notes, possible modifications

In a large hypermedia system, the solution of the set of equations describing a hypermedia network may be too time consuming, particularly since it would have to be performed for each outgoing link separately. We should thus consider shortcuts and suboptimal solutions and investigate their performance with respect to the optimal solution, both in terms of time and in terms of results.

One possible simplification is to precalculate the derived values of individual nodes within the context of the whole space rather than individual possible subspace, store these values in the nodes, and rank links according to the precalculated node values rather than from the increase of knowledge resulting from the traversal of individual links. In other words, the link to the most valuable node will be ranked first, and so on.

Another possible strat[illegible] to use a measure that does not require the solution of a set of simultaneous equations. The simplest approach is to replace derived value by intrinsic values. If we dispense with distinguishing between individual nodes and links by assigning all of them the same intrinsic value, the calculation is reduced to a measure of the density of interconnections. The advantage of this approach is that the values are calculated from a set of mutually independent formulae rather than from a set of linear equations and this requires a minimum amount of calculations. The disadvantage is that this does not capture the topology of the hypermedia space as thoroughly as our original model. This could be corrected by including second- and higher-order links by considering values of nodes reachable in two or three steps but weighing their contribution less than for immediately accessible nodes. This compromise requires more calculations but fewer than the solution of a set of linear equations, and its power of capturing the topology of the space is weaker than our original approach but stronger than for the most simplified approach.

BEST COPY AVAILABLE

An axiomatic approach

Our discussion raises the question of the properties that a reasonable measure of information in a hypermedia system should have. We propose the following:

- Adding a new fact or a new link with a positive value should increase the total value of the information space. Adding a fact or a link with a negative value should decrease the total value.
- If the connection topology of node A is equivalent to that of node B, then nodes A and B should have the same value.
- If node A is connected to more nodes with positive values than node B and everything else is equal, then node A is more valuable than node B, i.e. its derived value is bigger.
- If the structures of links attached to nodes A and B are the same but nodes attached to A have greater value than nodes attached to B, then node A is more valuable than node B.
- If a user has visited all nodes but has not traversed all links, traversing a link should increase his knowledge if links are valuable in themselves.

At the time of writing, we cannot prove that our proposed approach satisfies all these requirements. It would be interesting to see if a set of conditions such as the above can be postulated as axioms and a measure derived from it axiomatically in the way in which Shannon derived his information theory (Shannon 1948).

An extension

We have already mentioned that the value of nodes and links depends on the context, the purpose for which the hypermedia system is being used. Most large hypermedia systems contain information on many topics and all nodes do not have the same relevance for different users. Consequently, a user interested in a particular subject or having a particular goal will want to obtain a ranking of available links that maximizes the **increase of knowledge about a particular topic** or a mixture of topics or perspectives rather than knowledge as such. (By the term "perspective" we want to capture goals of users who are, for example, interested in rare information and similar goals that cannot be treated as topics.) This can be handled by replacing **scalar** values with **vector** values so that each element of each vector reflects the value of the node or link within a certain category of knowledge or perspective. As an example, component 1 may reflect the value of a fact for mathematicians, component 2 its importance for

ERIC

physicists, component 3 may be a measure of the uniqueness or strangeness of the node, and so on.

Corresponding to the vectorization of node and link values, the profile of each user's current interests needs to include a vector representing the relative weights of individual interests at a given moment. In the example given above, the first component will reflect the user's interest in mathematics, the second his interest in physics, and so on. To obtain the knowledge value of a hypermedia space for a particular interest, and to obtain the corresponding ranking of available nodes, the formulae given in the preceding sections must be converted to scalar products of vector quantities. The remaining reasoning remains unchanged.

Conclusion

We have addressed selection of links in large hypermedia systems because this problem will become important in large hypermedia systems of the future. We converted the problem into the question of quantitative evaluation of information present in a hypermedia system and discussed some preliminary proposals how to approach this problem.

We suggested to calculate the value of information on the basis of "intrinsic" and "derived" values of individual nodes and links. While we are able to formulate the basic principles and define several strategies for calculation of derived values, we cannot offer much advice on the calculation of intrinsic values.

After having presented our basic proposal, we pointed out the run-time cost of its implementation. We then suggested modifications that minimize implementation requirements but do not seem to capture the topology of the information space as well. In this context, we mentioned the need to formulate criteria (axioms) that a quantitative measure of knowledge should satisfy and noted that we are presently unable to state whether or not our most general approach meets all these conditions or how it should be modified so that it does. We are now working on these questions and attempting to obtain a better understanding of the behavior of the proposed approaches on representative hypermedia topologies.

It is worth noting that the problem that we are addressing could have applications beyond those of ranking links in a "passive" hypermedia system. One of them is to support the presentation of information to a student using an intelligent tutoring system. In this context, there are two networks, one being a model of the discipline and the other being the model of the student in which misconceptions are represented by nodes with negative values.

References

Botafogo A.R., Shneiderman B. (1992) Identifying Aggregates in Hypertext Structures, Third ACM Conference on Hypertext, San Antonio, Texas, December 15-18, 1992, pp. 63-74.

Shannon C.E. (1948) A Mathematical Theory of Communication. Bell Syst. Tech. J., vol. 27, 379-423, 623-657.

Tomek I., Maurer H. (1992) Helping the User to Select a Link, Hypermedia, vol. 4, no. 2, 111-122.

ERIC Full Text Provided by ERIC

SOLE - Smalltalk Open Learning Environment

Ivan Tomek
Jodrey School of Computer Science
Acadia University
Wolfville, Nova Scotia, Canada, B0P 1X0
e-mail: ivan.tomek.acadiau.ca

SOLE (Smalltalk Open Learning Environment) is a software environment for experimentation with computer applications in learning and for development of computer based learning material. Because SOLE is written in Smalltalk-80, it is highly portable across a number of popular platforms and fully extendible, allowing rather easy addition of new facilities and tools.

After outlining the motivation for SOLE, we explain its design philosophy and implementation principles, and describe the major tool now available in SOLE - the Book.

Introduction - the need for a portable extendible testbed

Application of computers to learning has a long history ranging from "electronic page turners" through animation, interactive learning facilities, and course management, to applications of artificial intelligence in the modeling of the interaction between a student, a teacher, and the teaching material. The multitude of acronyms created along the way (we will use the term Computer Based Learning Environments (CBLE) to group all approaches) to some extent witnesses the disillusionment with highly advertised movements that failed to produced the promised results, and the perceived need to disassociate new work from discredited acronyms. It also reflects the variety of approaches taken by different workers in this application of computers (Frasson 92, Tomek 92).

The growing interest in CBLE and the increasing variety of approaches point to the need for a widely available and transportable facility containing all essential tools to which new tools can be easily added. A facility with both of these attributes does not exists at present and our goal in designing SOLE, the Smalltalk Open Learning Environment, has been to attempt to correct this situation. When its most important components are implemented, SOLE

BEST COPY AVAILABLE

will be a software lab containing most of the tools needed to explore new ideas and new approaches. To encourage this sharing of ideas and tools on a broad scale and to make independent additions as easy as possible, we use Smalltalk-80 as the implementation language because of its unmatched portability and ease of code development and modification. In the following, we describe the SOLE philosophy and design principles, restricting ourselves to already completed work.

SOLE philosophy

In developing SOLE, our first goal has been to formulate a broad overall design philosophy to allow the desired easy extensibility and to provide a natural framework for authors and students. Our second goal was to develop the skeleton of the environment, and the third to implement the major tools that we consider essential for further work.

In formulating our overall aim, we opted for a natural metaphore that allows creation, access, and combination of multiple learning materials. The selected metaphore is a conventional learning environment such as a library containing information about learning materials (catalogs), the learning materials themselves (mainly books), information about users, facilities allowing users to access existing material and create new material, and a mechanism allowing tutoring with the use of the available learning material. In other words, SOLE is intended to provide the means to access and create material needed for learning, and to use it in ways in which instructors and educational administrators would use a conventional library.

In addition to the "public library" model, we recognize that learning material can also be purchased and treated as a personal possession rather than as a protected public resource. To make this possible, the SOLE public learning environment model is complemented by means to access learning material directly, without the need to deploy full SOLE. Whereas learning material used within SOLE is protected from the user in ways in which an ideal library would protect its holdings, individual owners of learning material can use their private copies any way they want. Figuratively speaking, the owner of a personal copy of a SOLE book can write over the pages, tear them out, paste in pages from other books, and so on. Since the focus of our current work is SOLE and the public library model of a learning environment, the rest of the article is limited to a discussion of the "public library" component of SOLE.

The structure of SOLE

SOLE is implemented as a class hierarchy that directly parallels our view of a library, a user, an author, and library material. The main components of the **SOLE** class in its present form are a catalog of holdings and a catalog of users whose selected features are explained below.

The catalog of holdings is an instance of class **Catalog**, a sorted collection of **BookInfo** objects, and the user catalog contains instances of **User** objects, a dictionary of users. The structure is as follows:

SOLE

- **catalog of BookInfo objects, each containing**
  - **authors**
  - **title**
  - **date**
  - **version**
  - **released**
  - **internal ID**
  - **access type**
  - **other information**
- **catalog of User objects, each containing**
  - **name**
  - **password**
  - **notepad**
  - **bookmarks**
  - **desktop**
    - **open books**
    - **current book**
  - **other information**

Each instance of class **BookInfo** contains information about a particular book, similar to a catalog entry in a library. Its main instance variables are the names of authors, the book's title, the date of its last modification or release, version (equivalent to edition number), internal ID for access of the actual book, and information about the type of permission required for its access (password-based mode of access requires the knowledge of the book's password,

BEST COPY AVAILABLE

ERIC

another of its instance variables).

A book may be in one of two states - under development or completed ("printed"). This distinction is implemented by a Boolean instance variable released whose value is false while the book is under development and true when the book is completed. While a "prerelease" version of a book may be modified by the author and by all users with proper access rights, a "released" book, like a printed book on the shelf, cannot be further modified, not even by its author, although the author can use the released version to prepare a new version. We will see later that although a released book is essentially a read-only document, individual users can append private annotations as desired and perform other modifications. However, these personal changes are restricted to personal use and do not affect the globally accessible released book.

Information about individual users is stored in instances of class **User**. The main instance variables of this class are the user's name and a password which provides access to the following personal information stored in the remaining **User** instance variables: the notepad, bookmarks, history, information about books marked by the user, and the desktop. We will now explain the less obvious of these variables in more detail.

- - The notepad is the user's private notepad, an ordered collection of notes. A **Note** is a two-element array consisting of the date when the note was created and the text of the note itself.
- - A bookmark is an object that can be created, annotated, positioned and repositioned within the book, moved to another book, and destroyed. Just as a conventional physical bookmark, it provides direct access to a location in a book marked by the user.
- Information about books accessed by the user and annotated or otherwise marked is stored in instance variable markedBooks. This object is a collection of IDs of books marked by the current user and provides access to files containing data about the various types of book "markings" made by the user. This information is directly associated with the user and separate from the book itself. When the user opens a book, this instance variable is checked: if the book is marked by the current user, the related files are opened and used to update the information about the public version of the book obtained from the global file.
- The desktop is an object containing a snapshot of the state in which the user last left SOLE. The main instance variables of class **Desktop** are openBooks and currentBook. Object openBooks contains a collection of instances of class **OpenBookInfo**, object currentBook points into openBooks. Class **OpenBookInfo** contains the id of a book currently open on the desktop and information about the user's position in it. When the

ERIC

user leaves SOLE and reenters it at a later time, the desktop opens in the same state in which it was left, with all previously open books opened at the same location and the last active book is active again.

The Book

We have already noted that our current work is limited to implementing the overall SOLE structure and the main components needed in typical CBLE work. Most authors would probably agree that the essential component of a library is a book. Naturally, modern libraries contain other items in addition to books but our view of a book is general enough to accommodate other types of library resources as well.

As we have already explained, each book is recorded in the catalog and selecting it in the catalog retrieves the book and displays its table of contents. Our implementation of a book is hierarchical, modeled after a conventional book: Each Book object has a tree structure whose internal nodes are "units" which correspond to chapters, sections, and subsections, and contain their names and information about their lower level components. The leaves of the Book tree are sequences of "pages". In implementation terms, a **Unit** is a dictionary whose keys are unit names and whose values are other units or page sequences.

Conceptually, each page consists of a style sheet and an ordered collection of frames. At present, a **StyleSheet** object is essentially a description of the page's background - the collection of items displayed on the screen when the page is activated and before any information about the contents of the page is examined. A style sheet may contain buttons, information about labels and colors, and other window items.

The fact that a style sheet is attached to each page does not mean that the author must create a style sheet for each new page. When a new page is created, its style is set to nil and remains so until explicitly changed by the author via the style sheet authoring facility. When the page is being displayed to a reader, its style is obtained on the basis of inheritance and the fact that all pages as well as units have a style sheet: If the page's style sheet is nil, SOLE looks at the style sheet attached to the parent unit of the page and if it is not nil, it is used to create the page. If the style sheet of the parent unit is also nil, the style sheet of its parent is examined, and so on. If none of the parents of the page has a style sheet, the system default style sheet is used. The default style is designed to satisfy most situations and to eliminate the need to create style sheets, while allowing style redefinition. This arrangement makes it possible to define a style for a book unit at any level and override it at page level if desired.

BEST COPY AVAILABLE

The concept of a style sheet is tied to class **StyleDictionary**, a dictionary whose keys are style names and whose values are instances of **StyleSheet.** When an author wants to attach a style to a unit or a page, he or she has two options: select one of the existing styles stored in the style dictionary, or create a new style. In creating a new style, two approaches are possible: The user can either select an existing style and edit it, or create a style sheet from scratch. Both approaches are interactive and use selection of widgets from a menu and direct manipulation of their visual representation on the screen.

The page object is not the bottom of the Book hierarchy since a page consists of an ordered collection of "frames". In our terminology, a **Frame** is a collection of all changes to the current display of an existing page that occur when the user indicates that he wants to proceed to the next state of the book material. A frame might add new widgets to the display or change an existing window item, for example by changing its position, size, color, or contents. A frame may also delete a widget currently on the screen.

The internal information that makes possible the display of a page as a sequence of frames consists of a **PageContents** object and a script. A **PageContents** object is a collection of all widgets that will ever be displayed on the current page. In other words, **PageContents** is the union of all window items created by all frames comprising the page. The **script** is used to develop a page on the screen one frame after another, and is captured in the instance variable <u>events</u>, an ordered collection of events, one set of events for each frame. Each set of events describes what happens in the corresponding frame by referring to objects in **PageContents**, specifying the desired action ("create", "change", "delete"), and supplying the necessary parameters such as position and size.

In addition to conventional book features, class **Book** also provides hypertextual linking. This is implemented via class **Link** whose instances allow one-to-many annotated links within a book or between different books. Link objects are essentially annotated location pointers whose interpretation is left to the objects pointed to. This makes the linking facility open-ended since to make a new type of window item linkable, one only needs to define a method interpreting the location pointer for objects in this particular class. A similar strategy also applies to annotations, making all window items annotatable. Finally, the distinction between the book's global links and a particular user's private links is made on the basis of the user's <u>markedBooks</u> object.

In addition to the Table of Contents and textual and graphical contents, conventional books often contain an Index and possibly a Glossary. These facilities are also available in the Book. A **Glossary** is a class containing a sorted collection of associations between terms and their definitions, and an **Index** is a sorted collection of terms and their locations in the book. An **Index** object is "active" which means that making a term-and- location selection from the Index window (containing two parallel selection lists listing currently available terms and locations) causes a transition to the indicated location in the Book. The user can modify both the **Glossary** and the **Index** objects without affecting the public Glossary and the Index attached to the Book via the markedBook instance variable in the **User** object explained above.

Book authoring principles

The Book authoring facilty is designed so that most authors will use it without requiring any programming. It contains a facility for easy creation of window items for pages and frames, presently providing text views for the display of text, code views for execution of Smalltalk code, selection lists, launchers with menus, various types of buttons, and graphics. The author can create, position, size, and fill these objects interactively after selecting the desired item type from a menu.

An unusual aspect of the Book is the way in which active window items are associated with their functions. In a typical authoring system, functions such as Next Page or Previous Page are either preprogrammed and assigned a fixed visual representation, or left to the author to program. The first approach makes it impossible to change the visual representation and implementation of the function while the second may require extensive programming. In SOLE, visible representation and function are separated.

Our goal in separating visible representations from functions is to eliminate as much programming as possible while retaining maximum customizability. The first goal is achieved by providing a variety of functions performing common operations such as Search, Open Index, Open Glossary, Go To Next Page, Go To Previous Page, and so on. However, we do not attach these functions to fixed visual representations but store them in a special class called **BookMethods**. Class **BookMethods** is simply a repository of all methods required by window items for communication with the user and the display.

When the author decides to implement a preprogrammed function on a page or on a style sheet, he or she selects the type of window item from a menu, defines its visual characteristics, and selects the function performed by the item by choosing the corresponding method from a menu. This makes it possible to implement, for example, the preprogrammed Next Page function using one of several kinds of buttons, label it with English text, French text, or an icon, or to use a pull down menu to implement it. This separation of function and

BEST COPY AVAILABLE

window item type and visual representation makes Book style fully customizable.

In the state in which the collection of functions is provided to the author, class **BookMethods** contains the standard functions required by a typical book as indicated above. However, if authors want to extend the range of functions, for example by creating a function to open a view capable of logic simulation (for a book on Digital Design), they can add the necessary methods and make them available through the selection list menu in the same way as built-in functions. Given the nature of the Smalltalk environment, new functions can thus be added relatively easily and without leaving SOLE and Smalltalk.

What we have said so far suggests that a Book is an object with limited means of interaction and this is true. In particular, Book support does not provide any tools to pose questions, examine answers, or allow alternative paths through pages. This may be an unusual limitation for an authoring system but it is intentional since real books do not have any such mechanisms either. Posing questions, checking answers, and making programmed decisions is an activity based on books but performed by other agents such as instructors teaching a course. This facility will be implemented separately as class **Course**.

Conclusion

At the time of writing, SOLE is limited to the SOLE framework (user and book catalogs and other features) and the Book. In our opinion, even this limited system provides a powerful facility of essential tools for computer applications to learning. After implementing the **Course** facility, SOLE will contain all the mundane features required for CBLE research. It will then be possible to produce conventional courseware material and courses based on existing Books, as well as create new types of books with new types of window items such as digital simulation. Since the **Course** facility will intercept all communication between SOLE and the user, it will then be possible to use SOLE as the basis for experimentation with more sophisticated problems such as Intelligent Tutoring Systems or natural language interfaces.

Since the implementation language of SOLE is Smalltalk-80 which is fully portable across numerous platforms, the completed SOLE will remove limits on cooperation between researchers using different hardware and software environments.

Acknowledgements

The design of SOLE has been influenced by numerous ideas and products including

the Xerox Analyst (Tomek 91), the Tigre (Tigre 91), IBM's PC StoryBoard (IBM 85) authoring system, and currently popular windowing environments.

Most of the Smalltalk programming to date was done by Earle Lowe and Oleg Verevka.

References

C. Frasson, G. Gauthier, G.I. McCalla (Eds.) (1992). Intelligent Tutoring Systems, Proceedings Second International Conference, ITS'92, Montreal, Canada, June 1992, Lecture Notes in Computer Science 608, Springer-Verlag.

IBM Corporation (1985). PC Storyboard, IBM Corporation, dept. M76, P.O.Box 2328, Menlo Park, California 94025.

Tigre Object Systems (1992). Tigre programming environment, Tigre Object Systems, 3004 Mission Street, Santa Cruz, California 95060.

I. Tomek, H. Maurer (1991) The Analyst as a starting point for a hypermedia system, Journal of Microcomputer Applications, 14, 139-160.

I. Tomek (1992). Computer Assisted Learning, Proceedings Fourth International Conference, ICCAL'92, Wolfville, Canada, June 1992, Lecture Notes in Computer Science 602, Springer-Verlag.

BEST COPY AVAILABLE

Stat Tutor: Integration of a CAL application into a one semester first statistics course

Dr. Peter J. Trushel
Department of Mathematics
Camosun College,
Victoria, B. C.
Canada V8P 5J2

Joanne Walker
Language Centre, University of Victoria
P.O. Box 3045
Victoria, B. C.
Canada V8W 3P4

Full Integration of CAL into courses and programs:

Today a large percentage of CAL applications are experimental or address a very specific aspect of a particular discipline area. It is common to see impressive CAL presentations which deal with only one very small portion of a particular course or program topic. In addition to this limitation CAL software often requires the use of expensive equipment that is still rarely available within small- to medium-sized educational institutions.

Because of their limited scope, experimental nature, and expense, CAL applications often are justifiably characterized as extensions, enhancements, supplements, accessories, augmentations, etc. to actual course or program presentations. CAL programs used in supporting roles represent important and useful educational tools, but because of their auxiliary and/or experimental nature they are the educational tools which are the most difficult to financially justify. Even in institutions where CAL has made inroads it is generally the first educational area to be cut in tight economic times.

If CAL is to become an essential part of the educational paradigms of smaller to medium institutions CAL applications must not be supplemental but must be fully integrated into courses and programs. They should also make use of the computing and AV equipment that is already available in the institution.

Creation of the **Stat Tutor**:

The authors have developed and classroom tested a CAL statistics application called the **Stat Tutor** that is fully integrated into a one semester statistics course. It is used in both lecture presentations and in out-of-class student explorations and applications. In addition the application requires only SE or Macintosh Classic computers and overhead projection pads that are already part

Stat Tutor: Integration of a CAL application into a one semester first statistics course of Camosun College's AV equipment.

The **Stat Tutor** is a Computer Assisted Learning application that has been integrated into a one-semester statistics course at Camosun College in Victoria, B.C. Written in SuperCard®, an authoring system produced by Silicon Beach Software of San Diego, California for use on a Macintosh computer, the **Stat Tutor** is a self-contained program incorporating the history and purpose of statistics, problem solving techniques, theory, examples, and problem sets. Since the program includes normal tables, t-tables, chi-square tables and a statistical calculator, students do not have to buy an expensive statistics textbook and a calculator for a one-semester course.

The calculator supports 26 variable names -- a through z, 50 student defined functions f1 through f50, and full statistical analysis of both frequency data and paired data. It provides graphs of any of the 50 functions and plots the regression line together with a scatter diagram. The resulting graphs are copied to the clipboard enabling students to copy any graph to the Macintosh scrapbook and from there to any application program. A Mathematics Geneva font allows expressions and functions to be entered in proper algebraic notation. Expressions and functions are then translated into SuperTalk functions for calculation. The Math Geneva font, developed by the authors, frees students from FORTRAN like data entry. For example sqrt(2*x^2 + 7*y^2) is entered as $\sqrt{(2 \cdot X^2 + 7 \cdot Y^2)}$. The calculator is only three text lines wide which allows most of the presentation screen to be viewed during calculations.

The statistical tables are also approximately three text lines wide and use scrolling fields to simulate the use of actual paper tables. This feature is very useful in classroom presentation as it allows the instructor to simulate the steps students use in table look ups.

The program is adaptable to any one-semester 100-level or 200-level course, including those that require a calculus prerequisite. Student workbooks that echo the in-class presentation are sold at the College bookstore at cost. At present, any student owning a Macintosh computer may have a copy of the **Stat Tutor**.

In the classroom, a Kodak projection pad connected to a Macintosh Classic computer projects the lesson onto a large screen. Essentially, the **Stat Tutor** used as an instructional tool is a computerized overhead projector presentation in which material is normally covered by a piece of paper then revealed as the topic is discussed. The student workbook reflects the projected, or on-screen, lesson. The workbook has blanks that are filled in by the students as the information is revealed on the computer screen. This approach obviates the need for students to take copious notes which is usually counter-productive because students generally lack the outlining skills necessary to take worthwhile notes and consequently waste a large portion of class time trying to decide what is important and how best to write it down. Students take home with them the lesson with the important information high-lighted in their own handwriting. In this way the information essential for students to remember, such as formulas and steps in theoretical development is emphasised.

The speed of presentation or review can be specifically geared to the audience. Review material or material a particular class finds easy to understand can be rapidly covered which leaves extra time for any topics which a class finds more difficult.

On the computer, the "blanks" are "buttons". Information is uncovered a blank at a time by clicking on a button with the mouse. By selecting Show from the pull-down Edit menu, all the information on the page is revealed. In a similar fashion all the material can be covered by selecting Cover. Show and Cover activities are useful when students want to try a problem again on their own after reviewing the steps, when students simply want to check if they have copied their notes correctly, or during class room presentations. The use of pull down-menus frees almost the entire computer screen for the presentation of information. This is particularly important when one of the low end small screen Macintoshes are used.

Another method of presentation involves animation which can effectively emphasize some types of lesson material. Examples of the use of animation in the Stat Tutor include the demonstration of the relationship between the sample space and a random variable, the generation of probability distributions from the values of a random variable, and the graphing of scatter points and the accompanying regression line.

The Stat Tutor uses minimal programming structure both in navigation and in presentation. Navigation is accomplished by selecting a chapter to view from a table of contents. Within a chapter a particular page is selected from the chapter index. From any page students can move to the next page, previous page, chapter index, or to the table of contents by the use of a pull down menu or the equivalent "hot keys". This structure allows students to move to any page of the computer book in no more than four steps.

Students have access to **Stat Tutor** on twelve Macintosh computers in the College library as well as two in the Math Lab. Students use the computer lab to work problems, to check answers to problems they have done at home, and to obtain notes that they missed in class. They review problem solutions by uncovering the solution step by step. Students are encouraged to work problems out using a pencil and paper and only uncover steps to check their own work. From any screen of the computer book students have access to the statistics calculator and statistical tables.

A mathematics tutor schedules a portion of her time in the computer lab. She shows students how to run the Stat Tutor program, answers questions about statistics topics, and also helps with other mathematical subjects. A very small amount of time is required for students to become totally proficient in running the Stat Tutor program.

Students have been involved in the project in an intimate and creative manner. The debugging of the Stat Tutor application was primarily accomplished through student critiques of the program. The initial problem set has been greatly expanded and enhanced by the use of problems generated from student team projects.

In order to expand and enhance the number of problems and applications in the Stats Tutor an idea stimulated by Dr. Ben Schneiderman's ICCAL '92 presentation "Engagement and Construction: Educational strategies for the post-TV era" was used. (Schneiderman, 1992)

In a class, students are broken into project teams of two per team. During a semester five projects are completed. The projects replace the five term test that are normally part of the course. Projects require the creation of a real world problem from each of the topics covered during the project period. Sources that problems may be taken from include world news, Canadian news, local news, and Camosun College publications. In order to avoid any copyright problems students are .old not to simply quote news sources verbatim, but to use the news source as the basis of the problem. Projects must contain clear and grammatically correct problem statements which include the news source and the names of the authors. Problem solutions must be clearly presented, grammatical, and mathematically correct.

Students are encouraged to attend class because a great deal of additional material is given during class time in response to student questions and comments. An initial worry was that giving students the ability to obtain notes without attending class might drastically decrease attendance; however, this has not turned out to be the case. Most students realise the advantages of the classroom presentations and of other students' questions and comments.

The problem sets produced by one class are used in the next class. Project problem sets are broken into five project areas instead of the usual textbook approach of putting problem sets at the end of each section. The project approach requires students to determine which technique is appropriate for the solution of each problem because the normal section clues are not available. Using projects in this manner helps students develop the ability to determine which methods are appropriate without knowing which section a problem comes from. This approach should reduce the number of frustrated students who express their frustration with comments like, "I don't understand how I failed that test. I could do all of the problems when I did my homework.".

Creation of their own problems requires students to really understand the material and to appreciate statistical applications in the real world. Since every project group is guaranteed at least a "Galley Proof" assessment, teams really value the written critiques provided. This is in sharp contrast to the typical response to comments made on tests which all too often students don't even read, or read only in a very superficial manner. In the project environment students must read and understand the comments if they are to correct their mistakes and get credit for a project.

Projects are graded as "Camera Ready", "Galley Proof", "Second Draft", "First Draft", and "Rough Draft". Projects must be re-submitted until they receive a grade of "Camera Ready" or "Galley Proof". Completed projects are combined into a problem set which gives the names of the creating project team members and becomes part of the problem sets for future classes. Students show a great deal of pride in seeing their names and their projects published in a book that will be used by other students.

The rich diversity of the problems students have generated came as a very pleasant surprise. One of the main reasons for this is probably the wide variety of individual student interests. Problems come from a much wider range of human endeavors than any one individual would ever have been able to create.

Evaluation of the **Stat Tutor**:

Perhaps the main disadvantage with this instructional paradigm is that the maximum number of teams an individual instructor can comfortably supervise seems to be about eight. Some initial experience with teams of three suggests they do not work nearly as well as teams of two. Teams of two honestly share the work load while in teams of 3 it is much easier for one member to piggyback on the other two members. A typical university transfer class at Camosun College has

ERIC

42 students so the Project approach has only been used with technology bridging students since their typical class size is about 16.

The **Stat Tutor** has been used at Camosun College in Math 116 and in Math 218. Math 116 has a prerequisite of grade 11 mathematics and satisfies the statistics requirements for various social science programs including the criminology program at Camosun College. Math 218 has a prerequisite of one semester of Calculus and satisfies the requirements of various university and college programs requiring statistics with a Calculus component.

In the future it would be interesting to compare pass and fail rates of classes which use **Stat Tutor** with those that use conventional teaching methods. Designing a proper statistical experiment must consider the effects of different instructors, different textbooks, different class rooms, etc.

Class attendance in classes using **Stat Tutor** has been very good. The potential problem of having students skip classes because they can get their notes in the library has not occurred.

Student reactions have generally been positive to the use of the **Stat Tutor.** Reactions to the project approach has been somewhat more mixed. Some students seem to feel that they work harder than they would in a normal testing environment. It is the opinion of the authors that while students may work harder they have a much better understanding of how statistics are used than students who have not had to consider which situations require the use of which statistical concepts.

Other members of the Camosun College Math department have been slow to use the **Stat Tutor**. Some of the instructors continue to resist the daily use of computers in the classroom. Instructors who are knowledgeable about computers often want to modify the material to better suit their teaching styles. However, once they realize the amount of time and effort that is required to create or modify a CAL tool that is used in every class period for an entire semester they soon return to the familiar standard blackboard techniques.

Reference

Schneiderman, B. (1992). Engagement and Construction: Educational strategies for the post-TV era. *Computer Assisted Learning, 192, 39-45.* Paper presented at the 4th International Computer Conference.

The Effects of Field Dependence/Independence and Computer Expertise on Learning Application Functions in a Graphical User Interface

JOHN G. TYLER
IBM Corporation, Multimedia Systems Design
1000 NW 51st Street Boca Raton, FL 33432

The advent of multimedia personal computing systems brings with it a wide array of challenges for the software designer, particularly with respect to the design of human-computer interface (HCI) features. For an industry still struggling to fully exploit the capabilities of advanced graphical displays in personal system graphical user interfaces (GUIs), managing the additional complexities of audio, image, and motion video elevates this design challenge to a new level.

Exploiting multimedia capabilities is particularly attractive for those applications whose main purpose is accessing and/or transmitting information to the user. For more general-purpose computer applications such as word processors, spreadsheets, graphics editors and databases, the impact of multimedia may appear to be peripheral. Yet there is potential to transform one common feature of nearly all computer applications: online help. Online help can and should provide the instructional resources that a new application user needs to climb the learning curve. Today, however, there are few applications where online help can achieve that objective.

Currently, application help is typically 100 percent text-based. Only recently have a few applications incorporated graphical content into their help panels. With multimedia features now available for most personal computers, online help may become the one essential function where all general-purpose applications can exploit multimedia capabilities. Before that can occur, however, a significant leap forward in online help design must be achieved. Design innovations must be driven by research, particularly in educational multimedia and hypermedia.

A research question of particular interest is whether the visual, pictorial display of application functions in online help can have a significant impact on the rate of learning application functions. More important than this question, which concerns the main effects of instructional treatment, is the question of individual differences in response to the instructional treatment. Specifically, do differing levels of computer competence or expertise affect learning in a multimedia help environment? Similarly, does a computer user's cognitive style affect their learning outcomes when learning from online help in a multimedia user interface?

To investigate such questions, a prototype multimedia help enabler was developed. This research project explores the design possibilities and consequences inherent in multimedia online help systems. The goal was to design and prototype a multimedia help enabler, and to understand some of the impacts that visually enhanced help would have on users -- those who increasingly rely on help, rather than manuals, to learn to use applications.

The Research Problem

Human-computer interfaces (HCI) provide an environment for the interaction between a human user and the highly dynamic operations of a computer. Graphical user interfaces (GUI), a variety of HCI, use pictorial as well as textual display elements to represent computer states and functions (Shneiderman, 1983; Horton, 1990). Different computer functions are understood to varying degrees by users who have learned computer operations. Learning to recognize and manipulate computer functions may require learning from information displayed online, in the form of system or application help messages. Online help messages typically consist of textual messages that explain a function or feature related to the help topic. Although functions are routinely represented in the GUI using non-verbal pictorial elements, online help messages rarely incorporate such message elements.

Instructional message design principles suggest that the modality of message elements used in performance situations should be the same as the modality employed during instruction (Fleming & Levie, 1978). Instructional design theory also suggests that the better a symbol system conveys the critical features of a concept or task, the more appropriate it is during instruction (Salomon, 1978). Reinforcing information presented verbally with appropriate visuals has been shown to result in significantly greater learning over presenting verbal messages alone (Fleming, 1987).

Cognitive style research has shown that field independent subjects identify and distinguish objects in a complex visual field more readily than field dependent subjects (Witkin et al., 1977). Such findings suggest that field independent computer users may learn more readily in an unfamiliar graphical user interface than would those who are more field dependent.

Comparative expertise research has shown that when introduced to new computer systems or applications, computer users differ widely in initial performance, depending primarily on the extent of their prior computer experience, and to a lesser degree on the difference between the type of interface encountered previously and the one being introduced (Whiteside et al., 1985). These findings confirm that where the potential for direct skill transfer exists, initial performance will be higher than where no prior experience exists. In novel user interfaces, however, the potential for skill transfer may be reduced to such an extent that some "experts" may learn even more slowly than "novices."

Prior research does not, however, provide clear answers to the following instructional design and human-computer interface design questions: Would either the user's level of field independence or level of computer expertise moderate the effectiveness of specific message designs for online help? In particular, would the level of field independence interact with varying levels of pictorial content in online help to influence performance on tasks in a GUI? Would the level of computer expertise interact with varying levels of pictorial content in online help to influence performance on tasks in a GUI? Would particular combinations of field dependence/independence and computer expertise interact in unique ways with varying levels of pictorial content in online help to influence performance on tasks in a GUI? These three questions embody the problems investigated in this study.

Graphical User Interfaces and Visual Learning

Pictorial or graphical user interfaces are rapidly replacing textual interfaces as the primary means for manipulating system and application functions, particularly on microcomputers used in office settings. By representing state and function using non-verbal, graphical elements, both static and dynamic, a graphical user interface can achieve greater expressive power than a purely textual user interface (Horton, 1990). By improving the fidelity of system function and state representation, a GUI should improve a computer user's ability to learn and use a wider range of functions than would be expected when a textual user interface is employed. Online help systems display documentation for computer operations and application functions via video display terminals (Horton, 1990).

Online help messages consist primarily of textual (visual-digital) message elements that verbally describe the help topic. Online help messages rarely incorporate pictorial (visual-iconic) elements that visually illustrate the help topic. This would be an appropriate design where the user interface is primarily verbal and textual in nature. Instructional message design principles suggest, however, that online documentation describing the use and manipulation of visual-iconic elements in a graphical user interface should itself contain visual-iconic elements that refer to and depict the computer operations and application functions.

As computer users interact with their systems using graphical, direct-manipulation techniques, learning computer functions shifts from a primarily verbal paradigm to an increasingly visual one. Strong verbal language skills cease to be the dominant aptitude for successful performance. Visual, non-verbal language skills should take on added value and have greater influence on concept attainment. This assumption follows from Paivio's dual-coding theory (Rieber and Kini, 1991).

Two aspects of dual-coding theory are significant when considering online help message design. First, verbal and visual cognitive mechanisms are interrelated; their learning effects are symbiotic. That is, information that is coded in both verbal and visual modes will be remembered more readily and more accurately than that which is coded only in one mode (Fleming, 1987). Dual-coding theory also suggests that visual stimuli will be encoded more frequently in both modes than will verbal stimuli, thus strengthening the value of visual, non-verbal stimuli for instruction.

Visual learning theory differentiates between presentations using static and dynamic graphical presentations. Graphics are believed to be effective attention-gaining devices (Rieber and Kini, 1991). When appropriately designed, graphics can enhance learning during computer-based instruction. Animated, or dynamic, graphical images are fundamentally different from static graphics. Animation in computer-based instruction involves rapidly updated computer screen displays, presenting the illusion of motion. Because computer states and functions are themselves dynamic, their pictorial or iconic representations in the graphical human-computer interface may be animated under specific conditions. Just as online help presentations should incorporate the pictorial elements found in the HCI, those elements should be animated if they are to be an effective representation of the interface during online instruction.

Individual Characteristics and Performance in HCI

Cognitive style: Accumulated evidence indicates significant variability of performance when users are first introduced to unfamiliar computer systems and applications (Whiteside et al., 1985; Pocius, 1991). Although the sources for such variance are too numerous to itemize, and while most sources have not been reliably measured, human-computer interface research is becoming increasingly focused on identifying the factors involved. For example, HCI research shows that ability, academic background, age, and previous computer experience are related to computer course performance (Pocius, 1991). From among several psychological characteristics that have been found to influence learning with computers, two have been selected for examination in this research: (1) cognitive style, and (2) computer expertise.

In their review of cognitive style research, Witkin et al. (1977) describe four essential characteristics of cognitive style. Cognitive style (1) refers to individual differences in how people perceive, solve problems, and learn; (2) is a pervasive dimension that influences personality, not only cognitive processes in a narrow sense; (3) is stable over time, although it may be easily manipulated or altered; and (4) is a bipolar dimension, where "each pole has adaptive value under specified circumstances, and so may be judged positively in relation to those circumstances" (p. 16). Witkin's research identified the cognitive style dimension of field dependence/independence (FDI), and has become the most widely investigated and applied to educational problems. His theory of cognitive style is not intended to narrowly categorize individuals as "field dependent" or "field independent". Rather, these are convenient, if somewhat misleading, labels for extremes of performance on perception tests (e.g., the Embedded Figures Test). In defining field dependence/independence, Witkin et al. (1977) assert:

> Because scores from any test of field dependence/independence form a continuous distribution, these labels reflect a tendency, in varying degrees of strength, toward one mode of perception or the other. There is no implication that there exist two distinct types of human beings." (p. 7)

In describing individuals' cognitive style, Witkin et al. consistently refer to relative, rather than absolute, characteristics. For example, "relatively field independent persons have been found more likely to impose structure spontaneously on stimulus material which lacks it" (1977, p. 9), and "the relatively field dependent person tends to adhere to the organization of the field as given" (p. 9). An individual's ability to perceive structure in a complex visual presentation, such as a computer application in a GUI, and to be able to correctly interact with that presentation, is expected to correlate positively with that individual's tendency to be field independent.

Computer expertise: Another factor that significantly affects performance in the human-computer interface is a user's level of computer expertise. In comparative expertise research, performance differences between novices and experts are analyzed, often in regard to how they perform in problem solving situations (Lesgold et al., 1990), or on other cognitive operations such as memory and perception (Aster & Clark, 1985). Whiteside et al. (1985) demonstrated consistent differences between expert and novice computer users' performance on tasks in unfamiliar and familiar user interfaces. The more familiar the user was with the style of interface presented, the higher their performance tended to be. User knowledge of computers, in most cases directly derived from experiences using them, is believed to be the most significant factor affecting computer user performance (Moran, 1981).

Applying ATI Research to Online Help Design

Aptitude X Treatment interaction (ATI) research is appropriate where the instructional design problem involves determining how specific elements of an instructional message might affect learning for certain individuals under certain task conditions (Clark & Salomon, 1986). In the design of online help messages, for users differing along dimensions of cognitive style and computer expertise, the instructional designer seeks to determine the level of information abstraction and the combination of media attributes that should be applied in order to maximize individual learner performance.

The general and specific effects that aptitudes such as field independence have across a variety of instructional treatments need to be better understood. ATI research techniques may be usefully applied to investigate such effects. Snow and Lohman (1984) describe the goal of theories of instructional treatment design: "There was a clear prescriptive goal for such a theory. It was the design of an adaptive instructional system ... [providing] alternative instructional treatments to fit the major differences in aptitude profiles among students" (p. 350).

One of the advantages of computer-based instruction, including the use of online help to provide instruction on the control of computer functions, is its potential to adapt each presentation to aptitude and ability characteristics of the learner. Where these characteristics can be reliably measured and understood with clarity and precision, it would be possible to apply instructional design principles so that the presentation can be adjusted to optimize the fit between the learner and the lesson. In order to design and validate adaptive instructional systems, principles defining the relationship between learner characteristics and specific instructional treatment variables must first be identified and their reliability established. This is a fundamental research objective of instructional designers developing intelligent tutoring systems (Wiggs & Perez, 1988; Perez & Seidel, 1990). This study pursues the validation of one such adaptive instructional design principle.

The goal of this research is to demonstrate principles of instructional design relevant to the design of online help in a graphical user interface. In addition, this research seeks to determine whether any aptitude X treatment interaction effects (between an individual user's cognitive style or level of computer expertise and the online help design) have a significant influence on learning and task performance in a GUI. Such research findings will help to advance the design of adaptive instructional computing systems.

Research Questions

The specific research questions addressed in this study are:

1. Is a computer user's cognitive style (field dependence/independence) significantly related to his or her performance on tasks in a GUI application where instruction is provided by online help?

2. Is a user's initial competence in GUI-based tasks, as a reflection of his or her level of computer expertise, significantly related to his or her overall performance on tasks in a GUI application where instruction is provided by online help?

3. Does the user's cognitive style (FDI) interact significantly with the presence of dynamic pictorial message content in online help to affect performance on tasks in a GUI application?

4. Does the user's level of initial competence interact significantly with the presence of dynamic pictorial message content in online help to affect performance on tasks in a GUI application?

5. When examined as combined traits, do levels of a user's cognitive style (FDI) and initial competence interact significantly with the presence of dynamic pictorial message content in online help to affect performance on tasks in a GUI application?

Methodology

These two independent variables, field independence and computer expertise -- as measured by the Group Embedded Figures Test (GEFT) and a computer experience survey instrument derived from the 1986 National Assessment of Educational Progress (NAEP) Computer Competence Test, respectively -- are analyzed as aptitude factors in an analysis of variance with performance on a training posttest as the dependent variable. Performance is assessed taking task difficulty, percentage of task complete, and time to task completion into account. Subjects were sampled from among volunteer business majors enrolled in a managerial accounting course. Those subjects who were at least .5 standard deviations away from the mean on each factor were selected to attend an individual training session. This sample population exhibited essentially normal distributions for both computer competence and field independence aptitude factors. Both aptitude measurement instruments were administered in a classroom at the university campus. Training was conducted at a corporate product evaluation center near the university.

After an introductory videotape lesson describing the computer system, the subjects completed an interactive online tutorial. All subjects then completed a three-task pretest covering system and application operations (e.g., create a folder, open a file, etc.) in the graphical user interface. The total performance score from these three tasks establishes a pretest performance score for each subject. Subjects were assigned to treatments from matched-pairs. Two subjects whose aptitude variables most closely matched were alternately assigned to either the Text-Only Help (TOH) treatment or the Text-with-Motion-Video (TMVH) treatment. The subjects then completed 12 lessons arranged as a sequence of tasks covering the operations of the system GUI and the graphical spreadsheet application. The tasks in these lessons were arranged in order of gradually increasing difficulty. After the 12 lessons were completed, a three-task posttest is administered, repeating the tasks and subtasks of the pretest.

In the experimental instructional treatment, dynamic pictorial message elements were displayed using a digital motion video playback feature integrated with the application help facility. The help text messages in both treatments were identical. To view the help text messages, the subject had to select the system or application help function. To view a video, in the TMVH treatment, the subject must first open the text help window, and then select a Video button within that help window. Video segments were produced to simply protray a given procedure or function. The video segments were dynamic, real-time visualizations of the help messages verbally presented in the help text windows. The principle of dual-coding was rigidly adhered to, insofar as the digital motion video presentation format made possible.

Results

A sample of university students enrolled as business majors were drawn from a pool of volunteers for computer training. The students were told that the training would introduce them to an advanced personal computer operating system (IBM Operating System/2 v2.0™) and a graphical spreadsheet application. The training would be held at a nearby corporate product evaluation center. The students were advised of the preliminary testing to be held on the university campus. 33 students attended these testing sessions during the Fall semester, 1992. At these sessions, students were given the 18-item Group Embedded Figures Test (GEFT) and a 94-item Computer Experience and Competence Survey (CECS). The latter instrument was developed specifically for this study by the author. These test sessions ran 1 hour, and all subjects completed both instruments.

Subjects from this sample were selected on the basis of extreme scores on GEFT and the CECS_E (subscale for computer experience). Subjects with scores falling 0.5 standard deviations from the mean on both scales were identified. 22 subjects thus selected were then matched on the basis of both GEFT and CECS_E scores, forming 11 pairs. One subject from each pair was assigned to the Text-Only Help (TOH) treatment, and the paired subject was assigned to the Text-plus-Motion-Video Help (TMVH) treatment. The 22 subjects were then scheduled for individual training sessions. Of these 22, 16 subjects attended the training and completed at least half of the instructional sequence (an instructional video tape, an interactive computer system tutorial program, plus six of twelve task-oriented CBI lessons). Six subjects who originally were scheduled did not appear for the training, or appeared but did not complete at least six lessons. This experimental mortality threatens internal validity of the study, but a series of t-tests show the final sample does not differ significantly from the original sample on either of the aptitude variables (GEFT or CECS_E).

Data for the final sample of 16 subjects who completed the training was analyzed using a 3-way (2*2*2) factorial analysis of variance with posttest performance set as the dependent variable. Pretest performance is taken into the model as a covariate to minimize the within group error terms. This is useful because of the very high correlation between the pretest and posttest scores (Stevens, 1990). As will often occur in randomized block designs, treatment mortality resulted in unequal cell sizes. This required the use of least squares means on which to base group comparisons. These were then used to test significance of first and second-order effects (main effects and interactions). Figure 1. below gives results for the ANOVA model POST= TREAT|CECS_E|GEFT PRE.

General Linear Models Procedure

Dependent Variable: POST

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	27475.31520	3434.41440	4.34	0.0138
Error	11	8697.30280	790.66389		
Corrected Total	19	36172.61800			

R-Square	C.V.	Root MSE	POST Mean
0.759561	21.74522	28.11875	129.3100

Source	DF	Type III SS	Mean Square	F Value	Pr > F
TREAT	1	29.00314	29.00314	0.04	0.8516
CECS_E	1	1313.19534	1313.19534	1.66	0.2239
GEFT	1	274.74067	274.74067	0.35	0.5675
PRE	1	16709.80637	16709.80637	21.13	0.0008
GEFT*TREAT	1	1852.22556	1852.22556	2.34	0.1541
GEFT*CECS_E	1	2112.52215	2112.52215	2.67	0.1304
CECS_E*TREAT	1	4657.78059	4657.78059	5.89	0.0336
GEFT*CECS_E*TREAT	1	80.53677	80.53677	0.10	0.7556

CCT_E	TREAT	POST LSMEAN
0	0	153.154445
0	1	123.790931
1	0	98.070227
1	1	132.696897

Figure 1. ANOVA Results (SAS GLM Procedure)

When accounting for the influence of pretest scores on posttest performance, the interaction of computer experience with treatment (an Aptitude X Treatment Interaction) is a significant effect with $\alpha=0.05$ ($F=5.89$, $p=0.0336$). The significance and direction of the interaction between computer expeience and treatment in this analysis demonstrates evidence of a relationship between prior computer experience and the relative effectiveness of multimedia-enhanced online information, for certain populations of users. Figure 2, below, depicts this interaction graphically, by plotting the posttest cell means (least squares means) for the computer experience by treatment interaction.

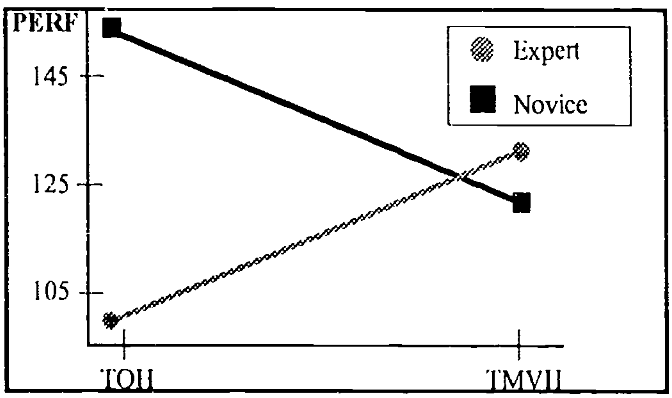


Figure 2. Aptitude X Treatment Interaction (TREAT X CECS_E)

The results of this study should reinforce the efforts of researchers to better understand the complex relationships between aspects of cognitive style and instructional message design variables. Novices appear to be negatively influenced by the presence of motion video presentations in the online information, while more expert users appear to benefit from the added dynamic visualization that motion video affords. Considering this interaction, it is reasonable to assert that the novice users readily engage in reading the text messages to learn new application and system function, but that the video images fail to reinforce these textual messages. This may be due to the novices relative lack of prior learned mental models with which they might comprehend the dynamic visualizations. More expert users have developed such mental models and may thus readily learn from these dynamic visualizations.

These findings should be viewed in a very tentative manner. The sample size in this study was very small, and the computer experience scale used is not well understood. Further studies are underway using this instrument, with alternate methodologies, to examine similar research questions. These results should, however, be of interest to researchers investigating other aspects of computer-based instruction design, particularly as applied to the design of online help and interactive system or application tutorial software.

Summary

Principles of instructional message design should be carefully applied to the design of online help in human-computer interfaces. Where those interfaces employ direct manipulation techniques, and rely on the use of pictorial symbols (icons) to represent system and application state and function, online help should also employ similar information representation symbologies. Learning should improve as the instructional conditions more closely resemble the criterion task performance conditions. Computer user aptitudes influence the way they learn and perform in the human-computer interface, resulting in detectable aptitude X treatment interactions. Detailed knowledge of such ATI effects may be useful to designers of adaptive instructional systems. This study identified one such relationship that exists among these design and user characteristics.

References

Aster, D. J. & Clark, R. E. (1985). Instructional software for users who differ in prior knowledge. *Performance and Instruction Journal, 24*(5), 13-15.

Burwell, L. B. (1991). The interaction of learning styles with learner control treatments in an interactive videodisc lesson. *Educational Technology, 31*(3), 37-43.

Canelos, J., Taylor, W., Dwyer, F., & Belland, J. (1988, January). Programmed instructional materials: A preliminary analysis. In *Proceedings of selected research papers presented at the annual meeting of the Association for Educational Communications and Technology.* New Orleans. (ERIC Document Reproduction Service No. ED 295 630).

Canino, C. & Cicchelli, T. (1988). Cognitive styles, computerized treatments on mathematics achievement and reaction to treatments. *Journal of Educational. Computing Research, 4*, 253-264.

Cathcart, W. G. (1990). Effects of LOGO instruction on cognitive style. *Journal of Educational Computing Research, 6*, 231-242.

Cavaiani, T. P. (1989). Cognitive style and diagnostic skills of student programmers. *Journal of Research on Computing in Education, 21*, 411-420.

Chiesi, H. L., Spilich, G. L., & Voss, J. F. (1979). Acquisition of domain-related information in relation to high and low domain knowledge. *Journal of Verbal Learning and Verbal Behavior, 18*, 257-273.

Clark, R. E. & Salomon, G. (1986). Media in teaching. In M. C. Wittrock (Ed.), *Handbook of research on teaching* (pp. 464-###). New York: Macmillan.

Educational Testing Service (1988). *A framework for assessing computer competence: Defining objectives.* Princeton, NJ: National Assessment of Educational Progress.

Federico, P. A. (1983). Changes in the cognitive components of achievement as students proceed through computer-managed instruction. *Journal of Computer-Based Instruction, 9*(4), 156-168.

Fleming, M. L. (1987). Displays and communication. In R. M. Gagne (Ed.), *Instructional technology: Foundations* (pp. 233-260). Hillsdale, NJ: Lawrence Erlbaum.

ERIC

Fleming, M. L. & Levie, W. H. (1978). *Instructional message design: Principles from the behavioral sciences.* Englewood Cliffs, NJ: Educational Technology.
Gregorc, A. F. (1984). Style as symptom: A phenomenological perspective. Theory Into Practice, 23(1), 51-55.
Horton, W. K. (1990). *Designing and writing online documentation: Helpfiles to hypertext.* New York: J. Wiley.
Lesgold, A. M., Gabrys, G., & Magone, M. (1990). *Cognitive and instructional theories of impasses in learning* (Final report). Pittsburg University, PA, Learning Research and Development Center. (ERIC Document Reproduction Service No. ED 317 578).
MacGregor, S. K., Shapiro, J. Z., & Niemiec, R. (1988). Effects of a computer-augmented learning environment on math achievement for students with differing cognitive style. *Journal of Educational Computing Research, 4,* 453-456.
Martin, M. A. (1983). Cognitive styles and their implications for computer-based instruction. Proceedings of the 24th International ADCIS Conference [Special issue]. *Journal of Computer-Based Instruction, 9,* 241-244.
Mestre, J. & Touger, J. (1989). Cognitive research: What's in it for physics teachers? *The Physics Teacher, 27,* 447-456.
Miller, A. (1987). Cognitive styles: An integrated model. *Educational Psychology, 7*(4), 251-268.
Moran, T. P. (1981). An applied psychology of the user. The psychology of human-computer interaction [Special issue]. *ACM Computing Surveys, 13*(1), 1-11.
Mykytyn, P. P. (1989). Group embedded figures test (GEFT): Individual differences, performance, and learning effects. *Educational and Psychological Measurement, 49,* 951-959.
Perez, R. S. & Seidel, R. J. (1990). Using artificial intelligence in education: Computer-based tools for instructional development. *Educational Technology, 30*(3), 51-58.
Pocius, K. E. (1991). Personality factors in human-computer interaction: A review of the literature. *Computers in Human Behavior, 7,* 103-135.
Post, P. E. (1987). The effect of field-independence/field-dependence on computer-assisted instruction achievement. *Journal of Industrial Teacher Education, 25*(1), 60-67.
Rieber, L. P. & Kini, A. S. (1991). Theoretical foundations of instructional applications of computer-generated animated visuals. *Journal of Computer-Based Instruction, 18*(3), 83-88.
Salomon, G. (1978). On the future of media research: No more full acceleration in neutral gear. *Educational Communications and Technology, 26,* 37-46.
Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *Computer, 16*(8), 57-69.
Sinatra, R. (1986). *Visual literacy connections to thinking, reading, and writing.* Springfield, IL: Charles C. Thomas.
Stevens, J. P. (1990). *Intermediate statistics: A modern approach.* Hillsdale, NJ: Lawrence Erlbaum.
Tall, D. & Thomas, M. (1989). Versatile learning and the computer. *Focus on Learning Problems in Mathematics, 11*(2), 117-125.
van der Veer, G. C. (1990). Operating systems in education: Mental models in relation to user interfaces. In A. Finkelstein, M. J. Tauber, & R. Traunmuller (Eds.), *Human factors in information systems analysis and design,* (pp. 223-241). Amsterdam: Elsevier.
van der Veer, G. C. & Wijk, R. (1990). Teaching a spreadsheet application: Visual-spatial metaphors in relation to spatial ability, and the effects of mental models. In P. Gorny & M. J. Tauber (Eds.), *Visualization in human-computer interaction,* (pp. 194-208). Berlin: Springer-Verlag.
van Merrienboer, J. J. G. (1988). Relationship between cognitive learning style and achievement in an introductory computer programming course. *Journal of Research on Computing in Education, 21,* 181-186.
van Merrienboer, J. J. G. (1990). Instructional strategies for teaching computer programming: Interactions with the cognitive style reflection-impulsivity. *Journal of Research on Computing in Education, 23,* 45-53.
Whiteside, J., Jones, S., Levy, P. S., & Wixon, D. (1985). User performance with command, menu, and iconic interfaces. *In Proceedings of CHI '85: Human Factors in Computing Systems* (pp. 185-191). Association for Computing Machinery.
Wiggs, C. L. & Perez, R. S. (1988). The use of knowledge acquisition in instructional design. *Computers in Human Behavior, 4,* 257-274.
Witkin, H. A., Moore, C. A., Goodenough, D. R. & Cox, P. W. (1977). Field-dependent and field-independent cognitive styles and their educational implications. *Review of Educational Research, 47*(1), 1-64.

Student Composition of Multimedia Documents: A Preliminary Study

Peri Weingrad
Kenneth E. Hay
Shari Jackson
Robert A. Boyle
Mark J. Guzdial
Elliot Soloway

of the
Highly Interactive Computing Environments
Research Group

Department of Electrical Engineering and Computer Sciences
The University of Michigan
Ann Arbor, Michigan U.S.A 48109

Introduction: Research and Theory of Multimedia in Education

Incorporating multimedia in the classroom is not a new idea. We have all seen audio/visual educational units on a variety of topics produced sometimes by teachers, but most of the time by professional or academic instructional designers. And most of us agree that education should take advantage of the power of these media. However, students traditionally have been fixed at the stationary and passive endpoint of these instructional units. This paper discusses the use of a new multimedia technology, the Macintosh™ computer software *MediaText*, which empowers students to be active participants in their education.

There is a long history of research on the effects of media on learning, much of it dealing with single media (for reviews of the literature, see Clark, 1983, Clark & Salomon, 1985, Pintrich, Cross, Kozma, & McKeachie, 1986, or Kozma, 1991). This research has focused the effectiveness of various media for learning and instruction. Extending this work to the frontier of multimedia research, researchers are exploring what functionalities of multimedia systems are beneficial to learning from instructional units. We diverge from this line of research following two insights in the recent educational research literature.

The first insight was a distinction made by Papert (1990) between instructionism and constructionism. This distinction forms the cornerstone of our beliefs about how multimedia should be used in the classroom. Instructionism is the notion that students are receptacles for the knowledge that the teacher or some other instructional medium imparts to them. Papert takes two steps away from this notion, first with constructivism and second with constructionism, where:

> "The word with the v expresses the theory that knowledge is built by the learner, not supplied by the teacher. The word with the n expresses the further idea that this happens especially felicitously when the learner is engaged in the construction of something external or at least sharable . . . a sand castle, a machine, a computer program, a book." (Papert, 1990, p.3)

The other insight on which our divergence is based is a distinction we attribute to Salomon, Perkins, and Globerson (1991), a distinction between the effects on student cognition *with technology* as opposed to effects *of technology*. In studying the effects *of technology*, researchers gather evaluation data after the student stops working with the technology. When studying the effects *with technology*, researchers evaluate the student/technology system to understand how the student changes.

We characterize mainstream educational research on multimedia as exploring issues related to the effects of technology within an instructionist paradigm. Our position, however, is that the most fruitful research of multimedia in education is in studying the effects with *technology* in a constructionist environment. In a constructionist environment, students work *with* media rather than being the receptors *of* it. Tools that allow the creation of multimedia artifacts to be as easy (or as difficult) as writing a book, building a sand castle or a machine would empower these students. To this end, the Highly Interactive Computing Environments (HiCE) group at the University of Michigan developed *MediaText*[1], a software multimedia composition tool. In brief, *MediaText* is to various media as a word processor is to text. It allows the user to create documents containing text, graphics, animations, sounds, and video. *MediaText* requires the student to be creative in the same way a word processor does in writing a paper, as well as providing students with choices unavailable before; that is, the ability to express themselves with images or sounds, and with text. The purpose of the study reported in this paper was to observe the uses students made of *MediaText*, and to describe the distinctive attributes of their multimedia compositions.

MediaText: A Multimedia Composition Tool

MediaText was intended to providing teachers and students with the expressive capabilities of multimedia and computers, but with none of the previously associated up front cost of learning a traditional programming language, or even HyperCard, to work with different media. The main features of the program seen in Figure 1, below, are as follows:

•<u>A standard word processor</u> comprises the left-hand three-quarters of the document window and allows for multiple styles, sizes and fonts.

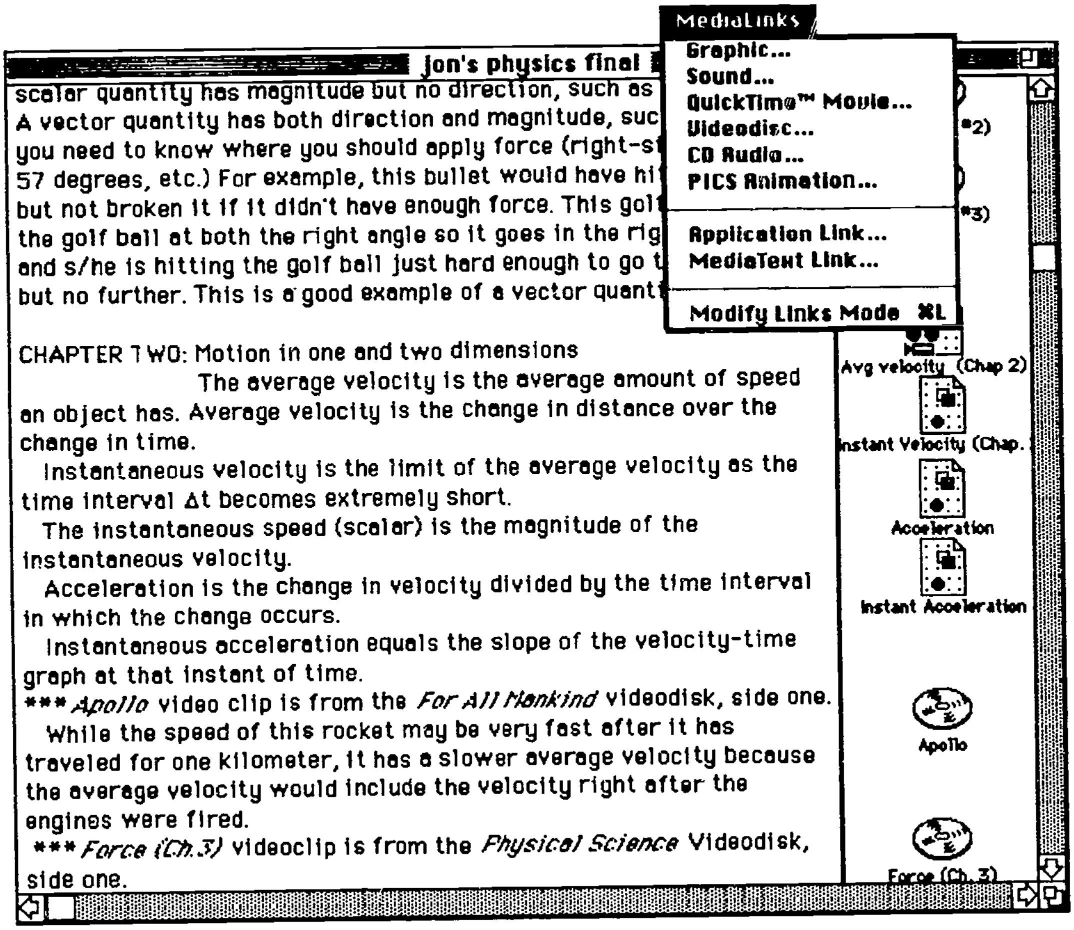
•<u>MediaLinks</u> are icons which represent the media incorporated in the current document. Clicking these icons plays back the MediaLink.

•<u>The Media Margin</u> on the right-hand side of the document window which contains the MediaLinks maintains the spatial relationship between the MediaLinks and text in the word processor while scrolling, so that MediaLinks remain next to the text where they have been placed.

•<u>The MediaLinks Menu</u> is a pull-down menu through which Media Workshops which contain tools for creating, selecting and/or editing MediaLinks that play graphics, animations, sounds, videodiscs, audio compact discs, and other documents.

- •<u>The Videodisc and Audio Compact Disc Workshops</u> incorporate controls for a computer-controlled videodisc player or CD-ROM drive wherein an author creates a MediaLinks by selecting starting and ending frame numbers or time code.
- •<u>The Graphic Workshop</u> is a built-in drawing editor with the capability to animate drawn objects simply by recording the author's motion when an object is moved across the screen.
- •<u>The Sound Workshop</u> allows the author either to digitize sound, or to import a previously created sound file.
- •<u>The QuickTime™ Movie and PICS Animation Workshops</u> link QuickTime™ Movies and cell-based animations of the PICS file type to a *MediaText* document.
- •<u>The Application and MediaText Link Workshops</u> links any other document (*MediaText* documents for hypertext capability) or application.

Figure 1: A Sample MediaText document



Description of the Study

MediaText was first used during the 1990-91 school year at Community High School in Ann Arbor, Michigan. Community is an alternative public high school where students are admitted to the school on a first-come, first-serve basis with no other entrance requirements. Typically a small proportion of the community's minority students elect to enter this school.

The computer lab in which the students in this study did their work was available to all students every afternoon from 2 to 5 PM and equipped with ten networked Macintosh II and SE computers, and a LaserWriter. These stations were connected either to videodisc players or to CD-ROM drives. Other equipment and materials available for student use included hand-held scanners, sound digitization hardware, a clip-art library, and videodiscs ranging from discs designed for classroom use to popular movies. Students used their own audio compact discs.

We studied the work of students who used *MediaText* to meet assignments in four classes conducted over the course of the school year: English Composition and Physics during the fall semester; Science and Society, and a course entitled "Multimedia: Art and Technology" during the winter semester.

English Composition

In the English Composition class, *MediaText* was used for two assignments by small groups of students volunteers working with a technical support person in the computer lab while the rest of the class worked on the same assignment with the regular classroom teacher.

The genre of the first assignment used with *MediaText* was "biography." The entire class devoted two weeks to the assignment; students under study took from three days to two weeks to complete their work. The students chose subjects for their biographies ranging from friends of the author to local television personalities and musicians. For the second assignment, writing a "how to" essay, the students working with *MediaText* chose topics such as "how to be annoying," "how to skip school," and "how to be a leaf."

Physics

In the Physics class, the class was divided in half, and two days of the week each half of the class worked in the computer lab with a technical support person. *MediaText* was used in this class first on an ongoing basis; students were asked to keep a "chronicle" or journal of what they had learned in class, illustrating this journal with MediaLinks. At the end of the fall semester, the Physics students were given the option of either taking the standard semester final exam, or creating a *MediaText* document that would discuss and/or demonstrate all the concepts covered during the semester. Students choosing the multimedia option were given a list of topics to cover and were required to use a variety of media to illustrate their final papers. These students were allowed to use the time the rest of the class spent reviewing for the in-class exam working on their papers, approximately eight class periods.

Science and Society

The focus of the Science and Society class was to study how scientific and technological improvements affect societal practices and vice versa. For an assignment on the development of some type transportation, six student volunteers were selected to work with *MediaText*.

Multimedia: Art and Technology

The course entitled "Multimedia: Art and Technology" was focused simply on using *MediaText*. This course was offered after school with three members of the research team as instructors. Nine students were chosen out of those who signed up on the basis of their schedules to ensure that they would be able to meet after school for two hours each week and that they had free access to the after school computer lab hours. Students in this class were assigned five *MediaText* projects over the course of the semester as described, in order, below.

- Introductory Assignment was an open-ended assignment designed to allow students to freely explore the media available to them.
- My Greatest Fear was an assignment wherein students were asked to create a document that would convey to their readers their feelings of fear.
- City assigned the students to report on a city of their own choice.
- Video Tape was an assignment wherein students were given a *MediaText* document containing a brief outline of a class discussion of the production of a videotape that would promote *MediaText* to an educational audience.
- Something Described was the final assignment where students created a *MediaText* document about some object or experience without showing it or naming it directly.

ERIC

Several of the students in this class produced documents for other classes or out of personal interest aside from the assigned projects. These documents were included in the total for this class and are labeled Independent Work.

Results

Our data set, summarized in Table 1, below, was comprised of sixty-two documents authored by twenty-two different individuals.

Table 1: AT and IC document typing by Class & Assignment

Class (*Assignment*)	n of students	n of documents	n of AT docs	n of IC docs
Composition (total)	7	7	7	0
Biography	5	5	5	-
How To	2	2	2	-
Multimedia: Art & Technology (total)	9	42	22	20
Introductory Assign.	8	11	8	3
My Greatest Fear	7	8	4	4
City	4	4	2	2
Videotape	5	6	1	4
Something Described	5	5	2	4
Independent Work	4	8	5	3
Physics (total)	7	11	5	6
Chronicle	4	5	4	1
Final	6	6	1	5
Science & Society (total)	2	2	2	0
Transportation	2	2	2	-
Totals for 4 classes *11 assignments*	**22***	**62**	**36**	**26**

*note that some several students were enrolled in more than one of these classes.

We performed a preliminary analysis of the data by reading the documents, and by counting text and MediaLink frequency. Out of this reading, two writing styles began to emerge. The first style of writing, which we call *Annotated Text* (AT), may be characterized as documents with fairly conventional-looking text, similar to a text-only essay. In AT documents, the text may stand

557

alone as a complete work and the Links provide parallel information. The second style of writing we refer to as *Integrated Composition* (IC). The "message" of an IC document tends to be distributed among the text and other media, and often the text was used to describe and explain the content of a MediaLink rather than the Link illustrating the text. Thus, the primary criterion for distinguishing between the two different document types was the extent to which a document's text "stood alone," that is, the degree to which the content of the text itself produced a coherent message.

The documents were then given to two coders. They proceeded to re-examine the documents, classifying each as AT or IC on the basis of this criterion. After classifying the documents as AT or IC, they compared each other's codings. Of the sixty-two documents, they both coded thirty as clearly of the AT style and twenty-four of the IC style. The eight remaining documents were somewhat ambiguous as to style; after some discussion, they classified two of these as IC documents and six as AT documents. Table 1, above, gives a breakdown per class and per assignment of the number of AT versus IC documents analyzed.

This coarse grain analysis, however, overlooks the individual differences within a document at the link level. During our reading of the documents we noted that students used a variety of techniques to relate the MediaLinks and text within their documents. As the next step in our analysis we began to explore the structural relationship between the MediaLinks and the accompanying text.

This analysis led to an emergent category scheme we refer to as Integration Techniques. We found that students had a range of different Integration Techniques which were either of a *explicit* or *implicit* nature. Explicit integration was demonstrated by the explicit reference to the link in the text. This was often, though not always, accompanied by a discussion of the content of the link. Implicit integration used any other means available to the author to alert the reader's attention to the relationship between a section of text and a MediaLink. These techniques are summarized in Table 2, below.

Table 2: Integration Techniques

Implicit Integration Techniques	**Explicit Integration Techniques**
JUXTAPOSITION is the technique of placing the Link near contentfully related text.	MENTION is where the name of a related link in the body of the text is mentioned, often with directions to activate that link.
TITLING a link in relation to the content of the Link and its related text was the next most often used implicit strategy.	DIRECTIVE is where the student directs actions the reader needed to perform to read the document (e.g., clicking, switching videodisks)
	SPECIAL TEXT is where the author uses parenthetical references, boldface text, indentation, asterisks or other characters to refer to the link.

•Juxtaposition Only two documents did not make use of this strategy. One of these, however, still presented the MediaLinks in the same content order as that of the text.

•Titling For the most part, students would name MediaLinks "appropriately" to their content, but there were occasional Link names with no apparent connection to the text of a document.

•Mentioning Students (infrequently) would discuss the content of the link and the relevance of the MediaLink to the content of the surrounding text (Note the discussion of the "Apollo" link in Figure 1, above).

•Directing Students often included directions to the reader on how to read their document. Directions ranged from alerting a reader when it was appropriate to play a Link to when to switch videodiscs.

Overall, how links were integrated by these techniques with the text varied widely between individual students. Often, students who wrote several documents would show different levels of integration between documents.

Interpretation

Our data indicate that less experience with *MediaText* results most often in AT style documents. The fact that all of the documents created by the students in the Composition class, the Science and Society class, and the chronicle assignments in the Physics class (which were produced early in the students' experiences with *MediaText*) were classified as AT bears this out. This is not to say that students who produced more than one document did not continue in this style. One student who produced several documents always wrote in the AT style, while others had documents of both types.

On the whole, however, we expected that with greater experience we would see more IC style documents. The trend appeared to hold in the Multimedia: Art and Technology class, our primary opportunity to observe trends of longer term usage of *MediaText*. Ignoring the independent work, which was performed at various times during the semester, we can see in Table 1, above, development from the initial assignment where there were eight AT documents and three IC documents to the final assignment where there were one AT documents and four IC documents. Several other factors need to be considered here, however. The first is the low number of documents for each assignment. Also, as some students did not finish later assignments, we cannot be certain about any trends we identify with these numbers. Thirdly, it should be noted that the focus of the class itself was to explore multimedia writing. However, the number of AT documents produced in this class indicates that more than experience is involved in determining the style in which a student will write.

Conclusions

Our work on student use of multimedia composition tools is still in its infancy, though we believe that more structured activities than were used in our study are probably needed to meet desired learning objectives through multimedia composition. For example, in a majority of the cases we looked at, students do not have extended discussions of the content of their links. If this were a desired outcome, the practice would need to be appropriately modeled for the students.

Despite the fact the study reported here does not allow us to make any firm knowledge claims with respect to the possible benefits of multimedia composition, there are several threads that merit further attention. For example, the issue of the difference between writing styles we have categorized as Annotated Text and Integrated Composition is suggestive for further work on the cognitive effects of student multimedia composition.

In particular, several aspects of Integrated Composition appear to relate to the issues of cognition and transfer mentioned in the introduction to this essay. By using multiple representations of a single concept in different media formats, students may be constructing a richer understanding of the concepts underlying these representations. And when a student discusses the relationship between the content of a link and the content of a section of text it may further their understanding. As students integrate MediaLinks and their text more explicitly, they may be thinking differently about the interconnections between the ideas expressed in the various media. Such a circumstance particularly speaks to the issues of transfer.

We suspect these cognitive processes can be facilitated by providing in the classroom a variety of resource materials that the students find meaningful outside of the classroom. In the Physics class, a student used frames from a videodisc collection of art to illustrate some of the concepts studied in class (e.g., a rosary bead in a painting having a mass of approximately one gram). Rather than looking for examples in science curriculum materials or science fiction movies, she and other students drew examples from *The Wizard of Oz* and *Who Framed Roger Rabbit?* to illustrate mechanical principles and laws discussed in the textual portion of their documents.

Finally, we are compelled to note that both the students and the teachers we worked with were excited about the opportunities for alternative assessments provided by a constructionist use of multimedia. Some of the students who chose to do the final paper in Physics noted that the notion of an in-class exam heavy in calculation problems was not only daunting, but it did not seem to address the way they understood the material. Through multimedia composition, the students were successfully trafficking in physics concepts where they might previously have been locked out by a more traditional approach.

References

Clark, R.E. & Salomon, G. (1985). Media In Teaching. In M.C. Wittrock (Ed.), *Handbook of Research on Teaching*. New York: Macmillan.

Clark, R.E. (1983). Reconsidering the research on learning from media. *Review of Educational Research, 53*, 445-460.

Kozma, R.B. (1991). Learning with Media. *Review of Educational Research, 61*, 179-211.

Papert, S. (1990) "Introduction by Seymour Papert" in *Constructionist Learning* edited by Harel, I. Boston: MIT Media Laboratory.

Pintrich, P.R., Cross, D.R., Kozma, R.B., & McKeachie, W.J. (1986). Instructional psychology. *Annual Review of Psychology, 37*, 611-51.

Salomon, G., Perkins, D.N., & Globerson, T. (1991). Partners in cognition: Extending human intelligence with intelligent technologies. *Educational Researcher, 20*(3), 2-9.

Wewant to give special thanks to the 1990-91 faculty and students of Ann Arbor's Community High School for their participation and partnership with the HiCE group, and to Jim Merz for continuing the development of the software, *MediaText*.

[1] The HiCE group is directed by Professor Elliot Soloway of the Department of Electrical Engineering and Computer Science, School of Engineering, at the University of Michigan. *MediaText* was originally conceived by Mark Guzdial, a graduate student at University of Michigan.

GEOMEDIA: A HYPERMEDIA SYSTEM ON EARTH SCIENCE TOPICS FOR MIDDLE SCHOOL CHILDREN

DENISE A. WILTSHIRE and CARMELO F. FERRIGNO
U.S. Geological Survey
801 National Center, Reston, Virginia 22092

GeoMedia is an earth science educational system that uses multimedia technology to teach middle school students about the hydrologic cycle, earthquakes, and maps. This multimedia system was developed by the U.S. Geological Survey (USGS) for use on Apple®[1] Macintosh® computer systems. Multimedia computer technology offers students exciting new possibilities for navigating through multiple and diverse layers of information. *GeoMedia* is also based on a newly emerging software programming technology known as hypermedia. Hypermedia techniques are used in *GeoMedia* to create a system that allows students to make associative links between a mix of information such as graphics, text, animations, and sound. Students are given the opportunity to interact with the information in ways that are not possible with traditional text books (Shneiderman and Kearsley, 1989, p. 72).

The *GeoMedia* Project began in the summer of 1991 to develop an earth science teaching tool that was based on the emerging hypermedia technology. *GeoMedia* was originally designed as an interactive presentation for students visiting U.S. Geological Survey Visitor Centers. However, to fulfill the many requests made by teachers to obtain copies of the software, the USGS decided to distribute the application on digital compact disc (CD ROM) for use in the classroom. Partially as a result of a favorable response from USGS employees, a second version of *GeoMedia* is being developed on global environmental change.

GeoMedia contains animations that show earth science processes such as the hydrologic cycle and plate tectonics. The three subject modules also include color images and sounds to further illustrate the topics. Glossaries and reading lists are included for each of the modules.

MacroMind® Director™ (version 3.1), produced by Macromedia, Inc., was the primary software package used to develop *GeoMedia*. The *GeoMedia* CD ROM contains the run-time version of the MacroMind Director software for viewing the system.

Design Production Elements

To meet the goal of designing an effective and innovative hypermedia application, an interdisciplinary design team was formed. The primary concern of the team was to communicate complex scientific processes using animations and terminology appropriate for grades 4 – 8. The team members had a mix of essential skills (Driscoll, 1991, p. 41) to perform the following critical tasks:

- Designing an overall identity;
- Identifying photographs, satellite imagery, illustrations, numerical data, and textual information;

[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.

- Designing the graphical user interface;
- Writing scripts and computer programs; and
- Testing and reviewing by earth scientists, teachers, and students.

At the start of the project, the team decided to produce *GeoMedia* using hypermedia technology as a basic design concept. In addition, the theme of a journey through earth science topics was adopted as an overall identity. The choices of platform and software were based on a previous prototype system that was developed to assess the effectiveness of hypermedia technology for communicating earth science topics (Wiltshire, 1991, p. 86). In addition to previous experience using Apple Macintosh computers for developing hypermedia applications, the existing base of these desktop computers in the education community contributed to the decision. The intuitive Macintosh interface and ease of use (Ess, 1991, p. 282) combined with the availability of commercial software for building hypermedia applications also affected the selection of a hardware platform. The selection of authoring software was also based on previous experience with MacroMind Director and the requirement for interactive animations (Rosenthal, 1992, p. 54).

The identification of data and processing of the information into the format for the targeted audience are the most time–consuming aspects of production. The data sets used in *GeoMedia* originated on a variety of computer platforms. Much of the information was in paper format; hence extensive scanning of maps and photographs was required.

Creation of the text for *GeoMedia* required extensive research into existing published material for children in grades 4 – 8 to determine the appropriate vocabulary and style for this age group. In addition, the graphical user interface dictated a limit of about 500 characters for the portion of the screen containing text.

The graphical user interface for *GeoMedia* was developed by the firm InterNetwork, Inc. The basis for selecting InterNetwork for this work was the firm's extensive experience in developing multimedia systems for science topics. One of the requirements by the *GeoMedia* project team was effective use of graphic design elements and typography. As noted in the information science literature, people have more difficulty reading from a screen than paper documents. However, through the use of an effective graphical user interface, on–screen readability is improved (Seyer, 1991, p. 109). The basic design gives the appearance of a video game in that most of the icons used for navigating through the application are located on a portion of the screen known as the control panel.

In addition to the narration during the animated sequences, sounds are used throughout *GeoMedia* to promote the theme of a video game. Originally cartoon voices were used for the narration. These voices were created by digitally manipulating an adult's voice. During the preliminary testing process, adult reviewers found the cartoon voice distracting. Children laughed at the sound of the narration, and the project team questioned how much of the information was being retained by the students. With the guidance of a professor of broadcasting, a young woman's voice was recorded for the narration to eliminate the cartoon sounds.

Scripting and programming were done primarily using MacroMind Director as noted above. An assortment of drawing packages were used for the creation of illustrations that formed the animations. Image enhancement software was also used in conjunction with high–resolution color scanners (flatbed and 35 mm slide scanners).

Extensive testing was required to ensure the scientific accuracy of the material presented and the reliability of the application. An advisory group of scientists from the U.S. Geological Survey conducted colleague reviews of *GeoMedia* to verify the accuracy of the information.

Local teachers were part of the review process as well. The teachers were asked to review the basic design and animations about halfway through the project. The teachers were also asked to assess the overall effectiveness of the hypermedia format as a viable method for teaching earth science topics. In addition, the teachers verified that the concepts were appropriately presented for the targeted grade levels. These reviews were conducted during a teacher's workshop sponsored by the USGS. The workshop consisted of a presentation on *GeoMedia* followed by a discussion. Each teacher was given an evaluation form as a guideline for assessing *GeoMedia* during the review process. The evaluation form was essentially a table of contents listing the major sections of the application. For each section, the form was supplemented with questions relating to suitability of the presented topics. This simple form was sufficient because the purpose of the the teacher's evaluation was to review the content and the suitability of the material for the intended audience. Finally, the beta version of *GeoMedia* was presented to 60 children in the sixth grade to test the effectiveness of the application. All of these levels of review contributed to the success of the application development.

Beta testing was performed by the project staff to ensure the functionality of the application. An elementary procedure was established along with a review form as guidelines for the beta testing process. Because each of the three modules in *GeoMedia* are identically structured, the procedures for testing the modules were also identical. The form was an outline consisting of a listing of links found in the main screen for each module. Project staff were instructed to test each of these links and provide comments as needed. Written instructions were provided for such items as proper configuration of the computer, checks for font inconsistencies and screen refresh problems, and other types of errors discovered in the development process.

Note that this procedure was a completely manual process and marginally satisfactory (in hindsight) for proper testing. For example, if an error such as an incorrect link was discovered, it was difficult to retrace the path taken through the application because there was no manual or automated procedure for recording this path. The record of this path can be referred to as an audit trail. Hence, for *GeoMedia2* manual or automated mechanism for recording audit trails may be developed. The initial use of this capability would be to determine if the links built into *GeoMedia2* are correctly programmed. However, there are other uses for audit trails (Misanchuk and Schwier, 1992, p. 360) such as determining the effects of taking different paths through the application and reader comprehension.

One of the more difficult aspects of software testing relates to the nonlinear arrangement of information that characterizes hypermedia applications. Because of the extensive layering of media and the many ways available to interact with *GeoMedia*, reviewers along with students may choose their own path through the application. This may result in each reviewer choosing different approaches to testing navigational functions, such as links between narrative sections of the application and a glossary. If for example, an incorrectly programmed link was discovered, it was difficult to duplicate the error because another tester might take a different path to the incorrect link and not find the error. Incorrect links of this type have been found in *GeoMedia* since it was published.

A need exists for standard methods to test applications that incorporate multimedia and hypermedia structures. Unfortunately, the authoring software used to create *GeoMedia* does not provide an automated capability to produce a schematic of all the links contained in the application. With a schematic of links, it may be possible to create a classification scheme for these links and then produce audit trails while testing the program. An automated procedure for generating audit trails can be programmed with the scripting language available with Macromind Director. With the current schedule for the production of *GeoMedia2*, it is more likely that a manual procedure will be implemented to generate a limited form of an audit trail.

Navigation Elements

The basic metaphor of *GeoMedia* is a tour through earth science topics—browsing through the multimedia system is similar to taking a journey through information (Figure 1). The *GeoMedia* graphical user interface allows students to navigate through the information by pointing at and clicking on icons with the computer mouse. The icons are located on a portion of the screen that resembles a control panel of an imaginary moving vehicle. The Tour Selections section of *GeoMedia* serves as the main table of contents. The student is given the option to select one of three earth science topics: Water Cycle, Earthquakes, or Understanding Maps. By clicking on icons in the shapes of diamonds, triangles, and squares, the student can navigate through the *GeoMedia* tour.

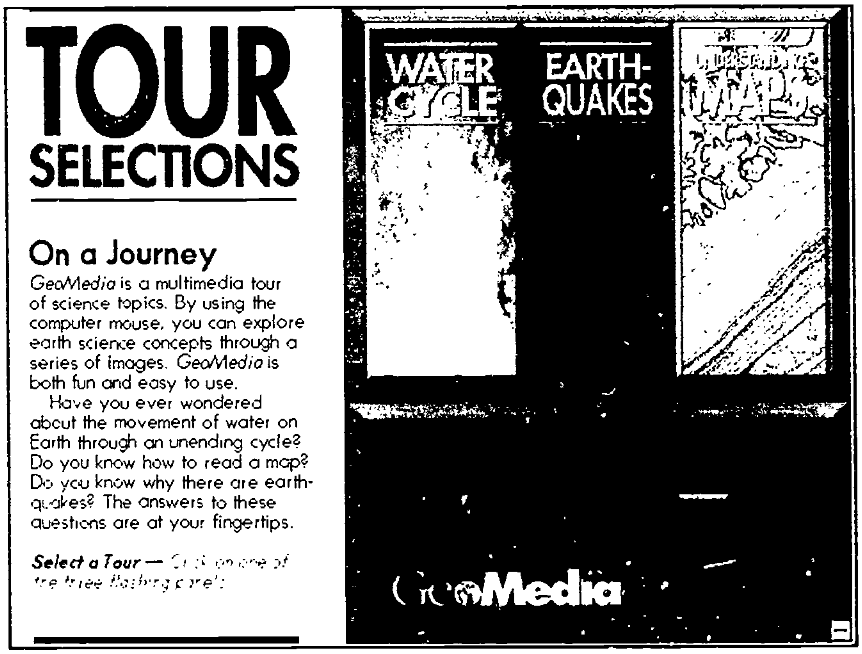


Figure 1. Computer screen from *GeoMedia* showing table of contents.

The Help option is always available while navigating through *GeoMedia*. The Help screen contains audible instructions in addition to traditional textual descriptions of functions. The audible instructions describe how to move forward and backward through *GeoMedia* and how to find definitions of words or additional earth science facts.

Each of the three tour selections contains four sections: Animation, Elements, Glossary, and Further Reading. Links between these sections allow students to plot their personal paths through the information. A general description of the tour selections is given below.

Subject Coverage

Water Cycle

This portion of *GeoMedia* shows the water cycle in an animated sequence. The animation explains the relationship between the water cycle elements such as evaporation, condensation,

ERIC

precipitation, and runoff. As noted above, links exist between the Animation and the Elements sections of the Water Cycle tour selection. For example, clicking on the cloud at any point during the animation links the reader to a more detailed description of evaporation in the Elements section.

Earthquakes

The earthquakes animation presents a series of images showing the forces that cause seismic activity. The images and audio portion of the system explain the relationship between plate tectonics and earthquake activity.

The Elements section contains text and images that describe the components of earthquakes in further detail than in the Animation section. In this section, students learn more about earth forces, plate tectonics, and monitoring seismic activity. Disaster photographs that show the destruction earthquakes can cause are part of the element on Pacific Epicenters which describes an area of frequent earthquake activity known as the Ring of Fire. Additional information on earthquakes is also available.

Understanding Maps

This part of the tour presents a series of animations that explain the concept of terrain elevation and the use of contour lines as map symbols. An animated three-dimensional model of a map explains the relationship between contour lines and elevation. A link exists between the three-dimensional model of a mountainous region in southern California and a series of satellite images shown in a perspective view near the same area.

The Elements section contains text and images that further describe the concept of map scale and the types of maps produced by the U.S. Geological Survey. Examples of planimetric, topographic, and thematic (geologic, hydrologic, and land use) maps are presented in this section. Also shown are scanned images of aerial photographs and satellite imagery.

Glossaries and Further Reading Lists

Throughout *GeoMedia*, icons that are embedded in the text, link the reader to the Glossary for the definition of the marked word. Each module contains a glossary that can also be opened directly from the table of contents.

Each of the three modules contains a Further Reading section to encourage children to learn more about the given topic by visiting their school or public libraries. The reading lists include references to periodicals, books, posters, and maps.

Teachers' Developer Version

The Teachers' Developer Version of *GeoMedia* is designed to give teachers and students the ability to create additional educational material on the earth sciences. The Teachers' Developer Version was created using HyperCard (version 2.1, distributed by Claris® Corporation).

The *GeoMedia* Teachers' Developer Version is comprised of a HyperCard application named *GeoDeveloper*. The application provides sample art work for screen designs and navigation icons to assist in creating a customized educational system. The control panel icons have associated HyperCard scripts and programming instructions for navigation. *GeoDeveloper* is a base to build upon when designing customized Hypercard applications. Scanned images, text, and sounds may be added to present additional information on the earth sciences.

ERIC

Preliminary User Responses

As of this writing, approximately 400 copies of *GeoMedia* have been distributed. With each distributed copy, a questionnaire is sent to provide the user with the opportunity to critique the application. About 25 educators returned the questionnaire and users have also contacted the project team by phone or in person.

From the preliminary user responses, three basic conclusions can be made. First, the application is appropriate for the targeted age group. Second, although *GeoMedia* was originally designed for middle school children, some college professors are using the application to teach the fundamentals of earth science. This substantiates the belief that well designed multimedia systems are appropriate for a wider range of grade levels than traditional textbooks (Flanders, 1992, p. 135). Third, users with less powerful Macintosh systems (LC, LC II, IIcx, etc.) indicated that at certain points the application seemed to run slowly. Many of these same users were running the program from the CD ROM instead of first copying the software (approximately 12 megabytes) to their hard disk. They either did not have sufficient hard disk space or they simply were not aware that this could be done even though this suggestion can be found in the user guide.

For *GeoMedia2*, several mechanisms are being tested for accelerating the execution time on slower Macintosh models. The first *GeoMedia* application, as previously stated, was developed for use in the USGS Visitor Centers. Each center is equipped with a reasonably fast Macintosh model, hence application speed was not an issue. For this reason, the animations are contained in several large files that load and execute more slowly on less powerful Macintosh systems. The animations, for *GeoMedia2*, are being divided into smaller–sized files, which will load and execute more quickly.

The use of QuickTime® software (Apple Computer, Inc.) is being investigated for *GeoMedia2*. QuickTime can be used to compress some of the large images that are part of the animations. This would reduce the overall size of the application and accelerate the display of the images. The use of QuickTime movies in *GeoMedia2* is also being examined. One of the limitations of QuickTime is the inability to create interactive links to other sections of the application without an additional developer's tool provided by the Apple Programmers and Developers Association.

Summary

The U.S. Geological Survey plans to continue developing prototype computer applications to promote earth science literacy among our Nation's school children. The use of computer and multimedia technology may inspire children who might not necessarily respond to traditional teaching methods. Multimedia uses all of the senses, creates an active instead of a passive learning experience, and can promote self expresssion and communication among students (Bruder, 1991, p. 23). In addition, computer technology continues to play an important role in most scientific research. Teaching earth science using computer technology as a tool offers new opportunities to give students a better understanding of scientific methods and investigative techniques, including data analysis.

Interdisciplinary teams are needed to build effective hypermedia applications. In addition to knowledge of the topic, essential skills are the ability to communicate complex scientific processes to nontechnical audiences, software scripting experience, and thorough understanding of the elements of graphic design. A need exists for comprehensive testing techniques to ensure software quality and reliability. Optimal software testing techniques should include the use of audit trails and related automated error detection schemes.

ERIC

References

Bruder, Isabelle. (1991). Multimedia: How it changes the way we teach and learn. *Electronic Learning*, 11(1), 22-26.

Driscoll, John. (1991). Designing the public interface, *in* Bearman, David, ed., *Hypermedia & interactivity in museums*, proceedings of an international conference. Pittsburgh, Penn., Archives and Museum Informatics, Technical Report no. 14, 35-45.

Ess, Charles. (1991). The pedagogy of computing: Hypermedia in the classroom, *in Hypertext '91*, ACM Conference on Hypertext, 3rd, San Antonio, Texas, December 1991, Proceedings. New York, N.Y., Association for Computing Machinery, 277-289.

Flanders, Bruce. (1992). Multimedia programs to reach an MTV generation. *American Libraries*, 23(2), 135-137.

Misanchuk, E. R., Schwier, R. A. (1992). Representing interactive multimedia and hypermedia audit trails. *Journal of Educational Multimedia and Hypermedia*, 1(3), 355-372.

Rosenthal, Steve. (1992). Author! Author! *Publish*, 7(9), 54-60.

Seyer, Philip. (1991). *Understanding hypertext–Concepts and applications*. Blue Ridge Summit, Penn., Windcrest Books, 263p.

Shneiderman, Ben, and Kearsley, Greg. (1989). *Hypertext hands-on!* Reading, Mass., Addison-Wesley Publishing Co., 165p.

Wiltshire, D.A. (1991). Arctic Data Inter*Active*–A hypermedia system, *in U.S. Geological Survey yearbook, fiscal year 1990*. Reston, Va., U.S. Geological Survey, 85-87.

ERIC

Visualisation of Semantic Relations in Hypertext Systems: A Set of Empirical Studies

ZHENGMAI ZHAO, TIM O'SHEA, PAT FUNG
Institute of Educational Technology, The Open University
Walton Hall, Milton Keynes, MK7 6AA, UK

INTRODUCTION

Although hypertext was initially invented as a new technology of structuring information rather than learning, the potential for applying such technology to learning are immediately apparent. The premise on which the most attractive potential is based is the high degree of learner control supported by hypertext. However, this has two sides. On the plus side, allowing learners to decide the pace and sequencing of navigation offers them opportunities to accommodate their individual differences and not only learn course content but also even to learn how to learn. On the minus side, it introduces the problem of cognitive overload (Conklin, 1987). Without sufficient guidance, learners, particularly novice learners, facing a number of choices about which links to follow and which to leave alone, experience a definite distraction, which degrades the effectiveness of learning. This might account for the fact that there are few evaluations of hypertext in education which have shown the strength of hypertext over other media in terms of overall learning outcomes. We believe that achieving a balance between the two sides of the learner control may be the key point in enhancing the use of hypertext for learning. Hypertext researchers and designers have been developing approaches to mitigating the problem of cognitive overload. Some approaches include guided tours, graphical browsers, computed links. In this paper we propose an approach to easing the problem of cognitive overload through attention to the basic design of hypertext learning systems.

This approach is closely related to the concept of semantic networks. Hypertext researchers and designers become aware of the analogy of hypertext structures to semantic networks (Conklin, 1987). Many of them believe that the characteristics of nodes and links offer hypertext the ability to mimic the semantic network (Jonassen, 1990). In fact, when Nelson first coined the term hypertext, he regarded hypertext as a means of describing a semantic network of knowledge (Nelson, 1965). Semantic networks have been claimed to be incorporated as a framework in a number of experimental hypertext systems, such as Thoth-II (Collier, 1987), and SemNet (Fairchild, Poltrock, & Furnas, 1988).

Research in the area of semantic networks is well documented (Quillian, 1968; Woods, 1975). Briefly, a semantic network is composed of nodes and links which represent, respectively, concepts and relations between concepts. The links in a semantic network are necessarily labelled explicitly to indicate such relations. A network without such labelling is a network, but not a semantic network.

Having taken inspiration from the issues described above and set them in the context of hypertext learning systems, we hypothesise that in a hypertext learning system, labelling links with link-types and letting the learner view them as far as possible might improve learning results. By "link-type", we mean the semantic relation between the link source and the link destination. Furthermore, we believe that this kind of labelling in making decisions about which links to follow and which to leave alone can lower the learner's cognitive overheads. This is because such labelling can serve as a guide to the learner by providing her with more specific reasons for the connections among nodes in the local hyperspace, or even

ERIC

in the global space. It is also because the provision of such labelling can make the conceptual model of the knowledge domain intuitively clearer to the learner. In this paper, a set of three empirical studies are reported which examined this hypothesis in different situations, using different methodologies. Initially, however, a number of specific terms important to the understanding of these studies, will be explained.

Vi. ble Link-Types (VLTs) As defined formerly, link-types refer to the semantic relation between two connected concepts (nodes). Visible link-types are the visualisation of such a relation. Various methods employed to visualise link-types are forms, colours and labels. The method chosen for our studies was to use labels based upon simple propositions. Clearly, formation of a proposition must be based upon both syntactic principles and semantic relation among involved concepts. What we are particularly interested in here is the semantic relations, the function of which is normally played by a verb. For example, in the proposition "John loves Sally" verb "loves" reflects a semantic relationship between concepts "John" and "Sally".

Embedded Semantic Net Hypertext Systems The current generation of hypertext can be divided into two categories in terms of whether or not a graphical overview of the network structure is used for navigation. Systems such as HyperTies and Guide where no structure diagrams are included fall into the first category and are referred to as embedded semantic net hypertext systems. In such systems, links are implemented by using either interactive buttons generally placed in a specific area of the document window or "hotpoints" embedded within context.

Explicit Semantic Net Hypertext Systems Systems like NoteCard and Intermedia belong to this category. In these, users are provided with diagrams showing a part or whole of the underlying structure network and rely heavily on the diagram for navigation. In explicit semantic net hypertext systems, the lines connecting nodes in the structural diagram are considered as an alternative representation of links in addition to interactive buttons and "hotpoints".

Structural Knowledge Structural knowledge, in the context of this paper, refers the knowledge of how concepts within a domain are interrelated. In hypertext systems which, as indicated in preceding section, can be conceived of as a knowledge representation, the structural knowledge is largely conveyed by means of such facilities as links, maps, and content tables.

Nodal Information This term refers to the basic elements of information held within individual nodes in hypertext.

STUDY 1

This study investigated the effects of visible link-types on goal-oriented learning in an embedded semantic net hypertext. It involved the evaluation of both learning outcomes and learning processes.

Method

Subjects Twenty adult volunteer subjects, who were staff or students of the Open University, took part in the study. They were randomly divided into two groups without regard to gender.

The Hypertext System The domain of this study was the search tree of data structures in computing. The hypertext system was implemented using HyperCard™, and was composed of 37 nodes (cards) and about 120 links, containing about 2000 words, and a dozen graphics and QuickTime™ animation. Links in the system were of two basic sorts: organisational and referential. The primary function of the organisational link was to build the hierarchical structure for the hypertext. In the system, such links as "start from the beginning", "go to the content table", and links from entries of a content table to their sections fell into this category. In general, the referential link is the kind of link that most clearly distinguishes hypertext from conventional text. Two nodes are associated to each other by a referential link based upon their semantic relationship. In our hypertext system, the referential links can be further divided into lexical links and textual links. The former connect regions of nodes to nodes, and the latter connect nodes to nodes.

Two different versions of this system were used as two different experimental treatments in the study. They were referred to as VLTs and NoVLTs. The two versions had the same contents, same structures, and same organisational links. In both versions, textual links were represented by buttons which were placed outside the main information space and lexical links were represented by the "hotpoint" which was a textual string with an underline. But the referential links, both textual and lexical, were labelled differently for two versions.

In VLTs, the textual referential link was labelled explicitly by using the semantic relation between the two nodes the link connected such as "has an example", "has contrast with", "is an analysis of", and so forth. In the case of the lexical referential link, since the link source was the part of text, its denotation was totally dependent on the context. To let the learner view the link-type, every "hotpoint" was attached with a small information palette. Once the learner moved the cursor over a "hotpoint", it was highlighted and its palette with the link-type information appeared at the bottom of the card.

In contrast to this, in NoVLTs, all textual referential links were labelled as "related information", and no link-type information palettes were attached to the "hotpoint".

Procedure The procedure included three steps. In the first step the subject was asked to do a pre-test, which consisted of two tasks: (1) filling in a printed self-assessment form which was designed to investigate the subject's prior knowledge of using a Macintosh machine, hypertext systems, and data structures in computing; (2) completing a hypertextual questionnaire including ten multiple-choice questions. Before starting the second task, subjects were told that the same multiple-choice questionnaire would be used in the post-test. In the next step, the subject was exposed to a specific version of the experimental hypertext system according to which group she had been assigned. When the subject felt her interaction with the system had been completed, she was in the final step, the post-test. The post-test also comprised two tasks: answering the same hypertextual questionnaire used in the pre-test, and filling in a printed satisfaction questionnaire, which was designed to determine the level of user satisfaction with the system.

A set of monitoring scripts were added to the system to record subjects' actions including the navigating track, the total time spent on browsing, the time allotted to each card, and the activation of referential links.

Results

Learning Results Table 1.1 presents the mean percentages and standard deviations of post-test scores for both the experimental group (VLTs) and the control group (NoVLTs). An examination of Table 1.1 revealed that mean post-test scores of subjects in VLTs group were ordered higher than mean scores of subjects in NoVLTs group. A one-factor ANOVA test was performed on the post-test scores to examine the reliability of this observation. The test yielded $F(1,18)=3.797$, the significant level $p=.067$. However, a further exploration of the data collected from the study led to applying a more precise statistical method. Even though the subjects were assigned at random to the experimental groups, an initial difference in learning prerequisites (equally weighted sum of two parts in the pre-test) in favour of the control group (NoVLTs) was observed. Moreover, a Pearson product-moment correlation test showed that subjects' learning prerequisites correlated positively and consistently with their performance in the post-test across both groups ($r=.25$, $r=.48$). A higher reliability of the observation or a lower significant level, in other words, was expected after the mean post-test scores of both groups had been adjusted or controlled for learning prerequisites.

Therefore, an analysis of covariance (ANCOVA) was decided to perform on the mean post-test scores, where the link group (VLTs vs. NoVLTs) was the between-group factor and learning prerequisite scores were utilised as a covariate. This more precise ANCOVA test yielded $F(1,17)=4.347$, the significant level $p=.052$, showing that subjects in VLTs group did indeed outperform subjects in NoVLTs group in the post-test.

Learning processes Observations of learners' behaviour in interacting with hypertext-based learning environments are the obvious methods to evaluate learning processes. The underlying rationale of use of such methods is the belief that human behaviour reflects cognitive processing (Marchionini, 1990). Statistical analysis on the data in relation to subjects' behaviour, however, did not show any significant differences between the two groups.

Table 1.1

Means (standard deviations) of percentage correct on post-test scores as a function of link group

link group	post-test
VLTs (n=10)	78 (13.98)
NoVLTs (n=10)	61 (23.78)

Discussion

The results of this study suggest that explicitly labelling links with semantic relations has a positive influence on goal-oriented learning in an embedded semantic net hypertext system. We strongly believe that the difference in learning outcomes should be reflected from learning processes as well. However, the behavioural observations do not provide any evidence in this aspect. The lack of difference might have been due to the modest scale of the experimental system or less sensitive measures for observing subjects' behavioural traces, or both.

STUDY 2

The purpose of this study was to examine if the proposed approach of visible link-types still held good in the context of the explicit semantic net hypertext system. As in Study 1 this examination was carried out by comparing both learning results and learning processes of two experimental treatments. In evaluating learning results, however, we looked particularly into two types of outcomes namely structural knowledge acquisition and specific nodal information gains. The same learning task (i.e. goal-oriented learning) as in Study 1 was adopted, but the scale of the knowledge domain was larger and the instruments for data collection were more sophisticated.

Method

Subjects Twenty seven paid subjects, who were first-year students at School of Computing & Mathematical Sciences, De Montfort University, Milton Keynes, took part in the study. They were randomly divided into two groups without regard to gender.

The Hypertext System A hypertext system was designed for this study implemented using HyperCard™. It was composed of 50 nodes (cards, floating and pop-up windows), about 150 links, containing about 3500 words, and a dozen graphics and QuickTime™ animation. Its domain was lists in computing data structures. This system was designed as an explicit semantic net hypertext where a global and a local diagram of underlying structure were available to users all the time.

The Global Diagram The purpose of the global diagram was to portray the whole hypertext structure. All primary nodes and links contained in the hypertext were present in the

ERIC

diagram, but were unlabelled and non-directional. Those nodes which had been browsed appeared on the screen as ticked and the node currently being investigated by the user was distinguished from the others by marking it in black. Users could expect through this diagram to obtain answers to the following types of questions at anytime during their navigation: "how large is the knowledge space I am dealing with? how interconnected is it? where am I now? how many nodes have I browsed? and how many nodes still remain untouched?"

The Local Diagram The local diagram appeared as a "zoom-in" view of the global diagram. Each node in the local diagram was labelled using the title of the concept represented by the node. As in the global diagram, the browsed nodes and the current node were marked distinctly. Users were expected to rely largely on this diagram for navigation. However, only those nodes which were directly connected to the current one could be approached. This could be fulfilled by clicking those small buttons attached to links (connecting lines) rather than by clicking nodes as in most hypertext systems. The reason for allowing jumping only to the nodes adjacent to the current one was based on the following concerns. Allowing jumping to any nodes just by clicking on them could make association links less meaningful so as to violate the knowledge structure imposed by hypertext authors. Of course, the full freedom of jumping in hypertext could be very useful for some kinds of users or at certain learning stages. The system provided two facilities to do that: the index facility and the history trail. The use of links as the "click area" rather than nodes themselves was intended to draw users' attention to link labels.

Two different versions of this system were used as two different experimental treatments in the study. The difference between the two versions appeared in the local diagram. In VLTs version, all links (connecting lines) in its local diagram were labelled by using semantic relations, which were represented by verL phrases such as "(be) a-type-of", "include", "compare-with", "(be) represented-by", and so on. Since relations had sequence, arrows were attached to connecting lines to indicate the sequence. In contrast, all links (connecting lines) in the local diagram of NoVLTs version were plain unlabelled lines.

Testing Materials A self-assessment questionnaire was designed to investigate subjects' prior knowledge and experience in using a Macintosh machine, hypertext systems, and computing data structures. In order to assess subjects' structural knowledge and nodal information gains independently, a two-part multiple-choice questionnaire was developed. Each part consisted of ten questions. Part 1 focused on relationships among important concepts contained in the hypertext while Part 2 placed the emphasis on the detailed information contained within individual nodes. Finally, a satisfaction questionnaire and an accompanying interview were designed to elicit each subject's views on the usability of the system.

Procedure Three steps were included. In the first step the subject was asked to do a pre-test, which consisted of two tasks: answering the self-assessment questionnaire, and completing the multiple-choice questionnaire. Subjects were informed before they started the second task that the same questionnaire would be used in the post-test. In the next step, the subject was exposed to a specific version of the experimental hypertext system according to which group she had been assigned. Subjects were required to verbalise their thoughts during their interaction with the hypertext. The whole process of their interaction was video-taped. When the subject felt her interaction had been completed, she was proceeded to the final step, the post-test. The post-test comprised three tasks: answering the same multiple-choice questionnaire used in the pre-test, filling in the satisfaction questionnaire, and being interviewed by the experiment administrator.

In addition to video-taping the browsing process, monitoring scripts in the stack were turned on to record subjects' actions including the navigating track, the total time spent on browsing, the time allotted to each card, and the type of jumping.

Results

Learning Results Table 2.1 presents the mean percentages and standard deviations of each part of the post-test scores (i.e. structural knowledge and specific nodal information) for both the experimental group (VLTs) and the control group (NoVLTs). An examination of Table 2.1 revealed that mean scores of NoVLTs group were ordered higher than mean scores of VLTs group regarding structural knowledge acquisition whereas the situation in the specific nodal information gains was just the opposite.

An ANCOVA test yielded $F(1, 24)=3.401, p=.078$ for the first part and $F(1,24)=.795, p=.381$ for the second part respectively. This indicates that even though subjects in NoVLTs group did better than subjects in VLTs group in understanding structural knowledge and subjects in VLTs group did better than subjects in NoVLTs group in acquiring nodal information, the differences were not however significant statistically.

Learning processes Again, statistical analysis on the data regarding subjects' behaviour did not show any significant differences between the two groups.

Table 2.1

Means (standard deviations) of percentage correct on post-test scores as a function of link group

link group	part1	part2
VLTs (n=14)	47.14 (18.99)	71.43 (24.13)
NoVLTs (n=13)	56.15 (18.50)	64.62 (22.95)

Discussion

Although results from this study do not appear to give credit to visible link-types, this might be due to the following two factors. Firstly, unlike the system used in Study 1, in this study, the system provides learners with not only the name of the current node but also names of all nodes adjacent to the current one. This may, in some cases, enable learners who have clear learning objectives to work out without other clues the appropriate semantic relationships of the links. In addition, reading the content included in the current node also helps this task. Such explanation was supported by interview data. When subjects of VLTs version were asked whether they paid attention to the link labels, most said that they did at the beginning but not afterwards because in some cases they could figure out the relationships without them. Secondly, because of the modest scale of the knowledge domain and the provision of structure diagrams, the phenomenon of cognitive overload expected to occur in navigating a "real" hypertext system was not observed in this study. This could have significantly obscured the treatment effects.

STUDY 3

Hypertext learning environments have been thought of as more suitable for open learning applications where no specific objectives are fixed since hypertext shares many of the characteristics that are integral to open learning. Such a learning task is different from the goal-oriented one used in former studies in that they involve different cognitive processing. The main purpose of this study was to investigate the effects of visible link-types on open learning in an explicit semantic net hypertext. In order to assess learning outcomes more extensively, besides the multiple-choice questionnaire, a teach-back test was introduced into this study, which was similar to the free-recall test. Both tests (multiple-choice and teach-back) consisted of two aspects namely structural knowledge acquisition and specific nodal information gains. In addition, the relationships between learners' spatial and verbal abilities and the impact of visible link-types on learning were also examined in this study.

ERIC

Method

Subjects and hypertext system Twenty four subjects similar to those in Study 2 took part in the study. They were randomly divided into two groups without regard to gender. Two hypertext versions which were identical to those in Study 2 were used as two different experimental treatments.

Testing Materials The same self-assessment questionnaire as that in Study 2 was used A two-part multiple-choice questionnaire was developed to assess subjects' structural knowledge and nodal information gains independently. Its first part consisted of ten questions and focused on relationships among important concepts contained in the hypertext. Its second part contained fifteen questions and placed the emphasis on the detailed information contained within individual nodes.

The result of the teach-back test was assessed using two scoring keys. The scoring key for the assessment of structural knowledge consisted of a set of propositions which covered the structural knowledge of the material. Each of these propositions was presented by a simple declarative sentence. Subjects' structural knowledge was measured by matching propositions in the scoring key with the idea units contained in subjects' teach-back tests. For each proposition that appeared in both the key and a subject's test, the subject received a score from 1 to 3 depending on the accuracy of the match. The scoring key for the assessment of nodal information gains comprised the catalogue of contents included in the material. Subjects' nodal information gains were measured by matching the catalogue with the description given by subjects in the tests. For each match, the subject was assigned a score from 1 to 3 depending on how detailed and accurate her description was. Reliability of this scoring method was established by having two people score the test. A significant positive association between the two sets of scores was found ($r=.91$).

In order to test subjects' spatial and verbal abilities respectively, GEFT (Group Embedded Figures Test) and DRVT (Delta Reading Vocabulary Test) were adopted in the study.

Procedure First of all, subjects were asked to complete the self-assessment questionnaire. Secondly, they were administered the DRVT (10 min). Following this, they were exposed to a specific version of the experimental hypertext system. Subjects were informed that on finishing browsing they would be tested on the information and they were also required to verbalise their thoughts during browsing. The whole browsing process was video-taped. When each subject felt her browsing had been completed, she was ready for post-test. This involved four tasks: teaching back what was learned from the hypertext, completing the multiple-choice questionnaire, filling in the satisfaction questionnaire, and being interviewed by the administrator. Finally, GEFT was carried out (18 min).

Results

Learning Results Table 3.1 presents the mean percentages and standard deviations of each part (i.e. structural knowledge vs. specific nodal information) in two post-test tasks (i.e. multiple-choice and teach-back) for both the experimental group (VLTs) and the control group (NoVLTs). Correlation analysis indicated that none of three independent variables (i.e. subjects' self-assessment of prior knowledge, DRVT scores, GEFT scores) was correlated highly and consistently with any dependant variables across both groups. Therefore, no independent variables were chosen as covariates to adjust the post-test scores.

An examination of Table 3.1 revealed that gaps between two groups on multiple-choice scores were so trivial that they could be ignored and that the differences on teach-back scores were quite large. To test the reliability of the second part of this observation, a t-test was conducted. The test yielded $t=1.81$, $p=.042$ (one-tail) for the first part and $t=1.166$, $p=.128$ (one-tail) for the second part respectively. It confirmed that subjects in VLTs group did significantly better than subjects in NoVLTs group in recalling the structural knowledge of the domain. Although subjects in VLTs group also did much better than those in NoVLTs group in recalling specific nodal information, the difference was not significant statistically.

Table 3.1

Means and standard deviations of percentage correct on post-test scores as a function of link group

link group	multi1	multi2	teach1	teach2
VLTs (n=14)	50.83 (9.96)	63.89 (18.08)	14.74 (8.41)	17.26 (9.42)
NoVLTs (n=13)	50.00 (24.86)	61.11 (20.66)	9.19 (6.15)	13.10 (8.03)

It was decided to explore further the relationships between (1) subjects' self-assessment of prior knowledge and their performance on the post-test; (2) subjects' DRVT scores and their performance on the post-test; and (3) subjects' GEFT scores and their performance on the post-test. In order to do so, subjects' self-assessment scores, DRVT scores and GEFT scores were used to form a second between-group variable in subsequent analysis. All subjects were respectively median-split into high and low self-assessment groups, high and low verbal ability groups, and high and low spatial ability groups.

Apart from the fact that mean scores of high self-assessed subjects in VLTs group were significantly higher than high self-assessed subjects in NoVLTs group in the aspect of structural knowledge for both tests (i.e. multiple-choice and teach-back) (*$t=1.898$, $p<.05$, one-tail; $t=1.967$, $p<.05$, one-tail*), no more significant differences were found.

Learning processes Statistical analysis on the data in relation to subjects' behaviour did not show any significant differences between the two groups.

Discussion

Since the two measures (teach-back and multiple-choice) used in this study varied in the degree to which cueing for the retrieval of information was provided to subjects, they involved different information processes. Few cues were supplied in the teach-back test so retrieval processes were highly operative in doing the test. In contrast, many cues were furnished in the multiple-choice test so that more encoding processes were involved in answering the questionnaire. The result of this study demonstrates that subjects in VLTs group achieved more than subjects in NoVLTs group in the teach-back test while no differences occurred between two groups in the multiple-choice test. It suggests that the visible link-types facilitate information retrieval processes but have less impact on encoding processes. It also suggests that visible link-types could help learners to construct the knowledge structure conveyed by the hypertext in their long-term memory.

As in Study 2, most subjects using VLTs version said they were not aware of paying particular attention to the link labels during their browsing. The results however seem to belie this. As does the fact that subjects in VLTs treatment were more satisfied with the system they used than were subjects in NoVLTs treatment.

Subjects with more prior knowledge were presumably more capable of taking advantage of the additional information provided by visible link-types than those with less prior knowledge. This might account for the fact that high self-assessed subjects in VLTs group significantly outperformed high self-assessed subjects in NoVLTs group in acquiring structure knowledge while low self-assessed subjects in VLTs group did not. However, this study reveals little with regard to the relationships between learners' spatial and verbal abilities and the impact of visible link-types on open learning.

GENERAL DISCUSSION

The results found from the current set of empirical studies appear to suggest that since no structure diagrams are provided in embedded semantic net hypertext systems, learners even with a clear learning objective could experience disorientation in choosing links to follow when the hypertext is relatively highly structured. Therefore, visualisation of link-types might help learners in both navigating and understanding the underlying knowledge. Provision of structure diagrams in explicit semantic net hypertext systems may overshadow the influence of visible link-types because the presence of structure diagrams itself, even without visible link-types, can ease learners' difficult in navigation. Bearing clear learning goals in their minds, learners may well be able in some cases to work out the correct semantic relationships assigned to most of links from the structure diagrams. However, when such a hypertext system is used for open learning, the effect of visible link-types again arises. Without fixed learning objectives, learners tend to feel more free to explore the hypertext. This point can be upheld by the following three statistics: the average time spent on browsing in Study 3 was thirty percent longer than that in Study 2 ($t=2.44$, $p=.009$, *one-tail*); the average number of cards browsed in Study 3 was thirteen percent larger than that in Study 2; and the subjects using VLTs version in Study 3 allocated more than twelve percent longer time on each card than subjects using same version in Study 2. This suggests the possibility that learners without a fixed learning goal in Study 3 paid more attention to visible link-types than learners goal-oriented in Study 2. This may provide an explanation for the higher level of achievement obtained by subjects using VLTs version in Study 3. Furthermore, experienced learners in open learning benefited more from visible link-types than inexperienced learners. The underlying reasons for these differences are not entirely clear.

In summary, the results of the current empirical studies show that labelling links explicitly with the semantic relations has a positive influence on learning outcomes. However, across the current three studies, the behavioural observations provide little evidence which supports significant results from the analysis of learning outcomes. This is may be because of the limitations of the current studies in such things as modest scales and less sensitive measures, or some other unknown factors. Seeking the answer to this question is undoubtedly a worthwhile research effort.

REFERENCES

Collier, G. (1987). Thoth-II: Hypertext with explicit semantics. *Hypertext'87*. University of North Carolina, Chaple Hill, North Carolina, 269-268.

Conklin, J. (1987). Hypertext: An introduction and survey. *Computer*, *20*(9), 1987, 17-41.

Fairchild, K., Poltrock, S. and Furnas, G. (1988). SemNet: Three dimensional graphic representation of large knowledge bases. In *Cognitive science and its application for human computer interaction* (ed. by R. Guideon), LEA, Hinsdale.

Jonassen, D. (1990). Semantic network elicitation: tools for structuring hypertext. In *Hypertext: State of the Art* (ed. by R. McAleese), Intellect Books, Oxford, 142-152.

Marchionini, G. (1990). Evaluating hypermedia-based learning. In *Designing Hypermedia for Learning* (ed. by D. Jonassen & H. Mandl). Heidelberg: Springer-Verlag, 355-373.

Nelson, T. (1965). A file structure for the complex, the changing, and the indeterminate. *Proceedings of ACM National Conference*, 84-100.

Quillian, M. R. (1968). Semantic memory. In *Semantic Information Processing* (ed. by M. Minsky), Cambridge, Mass: MIT Press.

Woods, W. A. (1975). What's in a link? In Bobrow, D. and Collins, A. (Eds.) *Representation and Understanding: Studies in Cognitive Science*. New York: Academic Press.

The Design of a Hypermedia Library Information System and its Evaluation for Educational Users

XIAOFENG JOHN ZHU

Musselman Library, Gettysburg College
Gettysburg, PA 17325-1493, USA

Hypermedia links text, graphics, image, sound, and video, and incorporates them into an accessible computer environment. With the development of information technology, hypermedia systems have become popular in recent years. Hypermedia is flexible and can integrate various types of information. It may have the potential to change the way we handle information storage and retrieval. Hypermedia is probably one of the ideal tools for processing and distributing library information. Yet, Hypermedia Library Information Systems are still rarely seen.

Work in a hypermedia application for library instruction and information services, called Library Information Multimedia Access System (LIMAS), was begun at Gettysburg College in 1990. LIMAS enables patrons to find library information and services in text and color graphics format. It can also interact with some Multimedia CD-ROM products such as Grolier Multimedia Encyclopedia, Microsoft bookshelf and Cinemania. Through the college campus network, LIMAS has access to the library online public access catalog (OPAC) and the college Campus Wide Information System (CWIS). LIMAS has been playing an important role in designing automated library instruction and information services at Gettysburg College, and it can be modified for use in other libraries.

LIMAS evaluation will focus on real users working with the operating hypermedia system. Computer monitoring and statistical analysis will be conducted to get a comprehensive and realistic picture of LIMAS usage and patrons' patterns and behavior on hypermedia systems.

INTRODUCTION

The concept of hypertext was introduced about half a century ago (Bush, 1945). Frisse (1988) provided an introductory article on hypertext, comparing it with regular text. In a comprehensive review on hypertext, Conklin (1987) not only summarized 20 hypertext systems, but also discussed the basic hypertext concepts and the advantages and disadvantages of hypertext.

In contrast to printed materials and traditional computer systems, hypertext uses nodes ("chunk" information) and links. It has the following main advantages (Nielsen 1990, p. 191):

- Non-sequential: it can handle both unstructured and structured data;
- User-oriented: it is relatively easy to learn and use;
- Multimedia: it can handle and integrate graphics, images, animation, audio, video, etc.
- Inexpensive: it is relatively easy to develop and update. It does not require sophisticated programming; therefore, not only experienced computer professionals but also self-taught computer users can develop hypermedia applications.

With the development of information technology and computer literacy, hypertext may

577

now be feasible, acceptable and affordable for many users. Hypermedia authoring software helps in designing a user interface and organizing and retrieving information in various electronic formats. This kind of software, such as HyperCard, Guide, and ToolBook, is available at an affordable cost. Many books on hypermedia software have also been published (Gillespie, 1990). A lot of hypermedia systems and applications have been developed (ACM, 1988; Byte, 1988; Conklin, 1987). In today's information society, hypermedia may play an important role.

THE NEEDS AND FEASIBILITY OF HYPERMEDIA LIBRARY INFORMATION SYSTEMS

Easy processing of and access to library information

Information technologies have brought new developments to library user education (Fleming, 1990). There is more and more information available in electronic formats such as text, computer graphics/images, sound, and video. Hypermedia systems, with links and multimedia features, may be used to organize, store, and retrieve various types of library information.

A library needs to promote its services by informing patrons as to how to find and use library materials and services. Traditionally, a library would prepare handouts, brochures, signs, or maps to help patrons. Librarians at the reference desk also take care of many patron inquiries. Although this method is useful, there are some accompanying limitations. For example, patrons may not like to read a lot of printed instructions in order to find one piece of information; it may be time-consuming; some patrons shy away from approaching reference librarians.

A well designed hypermedia system should overcome some of these limitations. It may enable patrons to search for information very quickly by links, cross references, indexes, etc., without reading other unrelated information. Since a hypermedia system can be made accessible on a network, patrons may use it from any place with a connected microcomputer. Such a system can greatly increase the effectiveness of the patrons' use of the library. Also, the users don't need to interact with a reference librarian for certain information needs.

Interfaces to other information systems

In a library environment, electronic information can be stored in different locations such as an OPAC, in-house information workstations, campus mainframes, and remote online sources. It can be retrieved through various systems such as CD-ROM systems, OPACs, Campus Wide Information Systems (CWIS), remote hosts, etc. With the development of library automation, networks, and online and CD-ROM database services, a library needs to deal with many information retrieval systems containing different types of information. It is difficult for libraries to integrate these systems and for patrons to use them without considerable help and/or training.

Hypermedia may overcome some of these limitations since it can be used effectively to design user interfaces to access various information sources. A hypermedia system may provide a good interface with online help for access to those systems, and also accommodate information in different formats. Therefore, it would promote better use of those retrieval systems.

Feasibility in design and development

1) Financial and human resources

A library generally has neither the funds nor human resources to develop extensive computer-based systems. Any application work must be accomplished with minimal funds and

ERIC

labor. Updates and modification of the system also should be within the library's capability. These criteria may be met by inexpensive hypermedia hardware and software, more computer-literate library staff, and more data available in electronic formats.

2) Inexpensive and popular platforms

The system should be based on economical and popular platforms such as IBM compatible microcomputers and Apple Macintoshes. This will make the system relatively easy and inexpensive to develop and operate. In addition, it is possible for users who own those machines to have remote access to the system if it is networked.

3) Compatibility with new technologies

The system should be easy to upgrade and able to adjust to new technology. Both the software and hardware the system is based on should be reliable and compatible with industry standards, so that the system will not be left behind by the development of new technologies.

Library-oriented systems

Hypermedia systems can be used in many places. The library is probably one of the ideal places. The following are a few examples.

The Computer Science Department of Indiana University of Pennsylvania used hypertext in a project that explored ways to improve subject access in an OPAC. A prototype system with 18,000 MARC records is up and running on a cluster of four NeXT workstations running Sybase. It is being expanded to 100,000 records (Micco, 1991).

The Book House project in Denmark utilizes an introduction screen that looks like a real library. Users select room icons to enter a reading room and then select graphical icons for different ways to find a book (Nielsen, 1990).

Neuman (1991) reported designing and evaluating MAJIK/1, a HyperCard program delivering basic and individualized library instruction. The formal evaluation was done with a pretest-posttest strategy and by means of naturalistic inquiry. Although students had problems navigating the program, the results suggested that MAJIK/1 met its goal and students liked to use it.

LIMAS SYSTEM DESIGN

Software and Hardware

The choice of a hypermedia development tool requires careful review of all options. Since the first widely used development tool, HyperCard, appeared on the Macintosh in 1987, many hypermedia applications have been developed on Macintosh, some on advanced workstations or minicomputer. However, many software and computer products for academic libraries have been based on IBM PC compatible microcomputers. For LIMAS to integrate with those products, a PC platform is more suitable. Fortunately, IBM PC compatibles have conquered the majority of the microcomputer market, and as a consequence, some hypermedia developmental tools have also become available for these machines. More tools have after Microsoft Windows 3.0 was introduced in May of 1990. Asymetrix's ToolBook is one of these tools.

The decision to use ToolBook in 1990 was based on many factors. ToolBook is very easy to use for hypermedia development, compared with other tools (Crass, 1990; Duncan, 1990). ToolBook uses color and multiple windows. It supports external Dynamic Link Libraries (DLLs) through which functions may be added to the environment, such as interfacing with full-motion video and digital sound equipment. It has a rich programming language that includes strong support for DDE links to other Windows applications.

Asymetrix has been working very closely with Microsoft to guarantee ToolBook's success. There is a runtime version of ToolBook bundled with each copy of Windows 3.0. A multimedia version of ToolBook is also available. The list price of ToolBook was only $495 in 1992. ToolBook is also bundled with some manufacturers' microcomputers for education users, such as IBM and Zenith, as well as some Multimedia PC (MPC) and upgrade kits, such as NEC MPC kit.

Therefore, it was decided to develop LIMAS for the PC running Windows.

System architecture and data structure

The Diagram in the Appendix shows LIMAS' logical structure. LIMAS design is based on a hierarchical structure with cross references and an index. It is divided into several modules (called books) in consideration of size, efficiency, and ease of maintenance and integration. The SYSTEM book contains frequently used programs and data for other books to share. The MAIN book contains major parts of LIMAS. The other books are for NEWSPAPER, PERIODICAL, STAFF INFORMATION, HELP, etc. Those will be loaded only when requested and removed right after the requests are executed, to conserve memory and speed up the system.

Each page (screen display) in a book is arranged according to hierarchical levels (see Appendix). Pages within the same level are stored together for easy browsing and modification.

Menu screens, index, cross references, and map location

LIMAS' menu screens are designed according to the hierarchical structure in the Appendix. On each screen, there are several buttons and arrows for selection. Clicking on any of them will change the button's color to confirm the selection, and then the corresponding page will appear on the screen.

On each menu there is usually a box that occupies the left 3/4 of the screen and a few buttons and arrows outside the box that occupy the right 1/4 of the screen. The buttons and arrows are identical on most of the screens. Their functions are described below:

BACK	-- go back to the previous screen (toggle between the current screen and the previous screen);
MAINMENU	-- go to Main Menu;
INDEX	-- go to Index;
HELP	-- go to the relevant help screen;
UP arrow	-- go to the menu screen one level higher;
LEFT/RIGHT arrows	-- go to the sibling menu or screen page

The buttons within the box are menu choices. They are located in the left 3/4 of the screen. Detailed information is also displayed inside the box whenever available. Those buttons (except the SEE ALSO buttons), along with the UP, LEFT, and RIGHT arrows function as a strict hierarchical tree-oriented navigator. From any screen (node), one can go to the parent (UP), a sibling (LEFT or RIGHT), or a child (the buttons within the box, except the SEE ALSO buttons).

The structure of buttons and arrows not only makes it very easy for browsing through related information (Schuerman and Peck, 1991), but also keeps users from getting lost in another screen without realizing how they got there. The relatively fixed locations of buttons, arrows, and information facilitate learning (Aspillaga, 1991).

Cross reference links are set up whenever appropriate. The links make it easier for patrons to find related information and unnecessary to duplicate information. Note that the UP, LEFT, and RIGHT arrows are disabled here. A user can only go back to the previous screen or go to the MAINMENU or Index. In this way, the system will not let the user jump to the middle of another section and so prevents the user from getting lost in an unfamiliar territory.

Each screen in the main book is indexed automatically with a specific term. The corresponding screen will be brought up after a term is clicked.

LIMAS has detailed color maps for each floor of the College Library. Most library materials or facilities have a LOCATION button associated with them. When selected, the exact location will flash on a map.

dBASE data for periodical and newspaper books

The PERIODICAL book contains about 2,500 titles from a dBASE file. By using ToolBook TBKDB3.DLL dynamic link library, all the data in the dBASE file is transferred into the PERIODICAL book. The menu of the book consists of titles sorted alphabetically. Each record can be accessed by clicking directly on the title or by the search button. The search adopts a binary method to speed the process. In addition, if a request is not found, LIMAS will show the alphabetically related titles from the menu. This also enables a user to type only the first few words of a title in order to search for it.

Access to other information systems

With appropriate multimedia hardware such as a sound card and a CD-ROM drive attached to a PC, LIMAS has been integrated with MPC products such as Grolier Multimedia Encyclopedia and Microsoft Bookshelf and Cinemania. It can also be designed as an interface for other information sources.

Linked with network software, LIMAS has the capability to access the CD-ROM databases, online library catalogs and campus-wide information systems through the campus computer network.

Guided tour and book marks

A common problem with some hypermedia systems is that there is no simple way for a user to browse through all or major parts of a system. LIMAS has a tour facility that allows a user to travel through LIMAS with a simple key press. (The space bar was selected for its central location and easiness to use.) Even people with no computer experience can use the tour easily by just pressing the space bar. The tour is designed to cover all areas with the depth-first algorithm on the hierarchy structure (see Appendix). Specific tours can be designed to cover selected areas for certain groups of users.

In addition, LIMAS has a book mark function which records and shows all areas the user has traveled by changing a button color. This function and the tour are integrated so that the areas traveled by a user are marked and the tour will not point to the marked areas unless selected by the user. Therefore, users always know which areas on the screen they have seen and they can take or leave the tour any time.

Help

HELP is implemented by pop up windows. The user can click on an italicized word or

ERIC

term from the help text, or click on the four buttons near the bottom of the HELP window. The screen choice TOPICS will show a list of help topics available, and the choice BACK will go to the help screen where the user first started HELP.

The HELP window appears in the middle of the screen. Therefore, the user can still see part of the original screen and all of the buttons outside the box. If any area outside the help box is clicked, the Help will exit automatically and the user will go to the place requested.

Current Status

The design and programming work for LIMAS have been completed. Most of the data in LIMAS is imported from the computer files available in the library. Frequent updates of the information LIMAS contains are necessary. Enhancement and modification are also under way. All work on LIMAS is done with the author and his student assistants' spare time.

LIMAS was tested and informally evaluated by several librarians from the college library and faculty members from the Computer Science Department. LIMAS was installed in the college library in August 1992. Hundreds of patrons have used the system since then. It has been praised by many users. It seems LIMAS has helped the library to better inform its patrons and has shown faculty and students the power of multimedia.

COMPUTER MONITORING AND EVALUATION OF LIMAS

Computer monitoring is a method which uses software to record and track system use. Penniman and Dominick (1980) presented a background survey of the existing state-of-the-art as it related to monitoring of information systems.

A variety of utilities or programs have been developed to log user interactions with information systems, such as macro definition utilities, specially designed terminate-and-stay-resident programs, vendor-supplied logging software, and customized versions of retrieval software to capture and time keystrokes and mouse moves. Some transaction logs also record system responses.

Rice and Borgman (1983) discussed the kinds of data available from computer-monitored information retrieval and communication systems. The data can be used for pattern analysis, error analysis, and time analysis. The advantages of this method included automated collection, the unobtrusive collection of computer-monitored data of known accuracy, full census and network data, longitudinal data, and automated experiments. Some of the disadvantages were extensive data management, ethics and privacy, and limited user information.

Very few studies have been done on an operating hypermedia system in a library, one of the most important places for information. "So far, however, relatively little detailed empirical research has been carried out of hypermedia in realistic use.... The field studies currently available make it difficult to draw specific conclusions about features of hypermedia systems. " (Waterworth and Chignell, 1989, p.207)

Computer monitoring is a major part in LIMAS evaluation study. This study will be considered as a field study in the sense that the system is operated in real setting (Gettysburg College Library) for a real users (the college community). The data sources to be monitored by computer are time, user screen moves, and destination. Both descriptive and inferential statistics as well as Markov Chain analysis will be adopted in the study. The data collection has started and will last about one year.

Very little is known about hypermedia's impact on users. With its sophisticated structure and features, hypermedia may put more cognitive loads on users. The questions explored by this study will be the following: 1) What patterns and behavior are exhibited by real patrons using an operating hypermedia system? 2) How can the patterns and behavior be characterized in terms of time, states, and tactics? 3) Are users ready to approach hypermedia? 4) How are specific features used in operating hypermedia systems?

This study will be one of the very few that examine user patterns and characteristics of an operating hypermedia system. In addition, it will be conducted in a library, an important place for information storage and retrieval, as well as learning and study. Thus, the findings of this study will certainly complement the large number of reports of laboratory studies, will help to provide a more complete picture of the field, and will contribute to research on user studies for hypermedia systems.

References

ACM. (1988). Special Issue: Hypertext. *Communications of the ACM*, 31(7), 816-895.

Aspillaga, M. (1991). Screen design: Location of information and its effects on learning. *Journal of Computer-Based Instruction*, 18(3), 89-92.

Bawden, D. (1990). *User-Oriented Evaluation of Information Systems and Services*. London: Gower.

Bush, V. (1945). As we may think. *Atlantic Monthly*, July, 101-108.

Byte. (1988). Hypertext. *Byte*, October, 234-244.

Conklin, J. (1987). Hypertext: An introduction and survey. *IEEE Computer*, September, 17-41.

Crass, D. (1990). ToolBook enables easy Windows 3.0 programming. *InfoWorld*, July 9, 63-64.

Duncan, R. (1990). Power programming: An introduction to building applications with ToolBook. *PC Magazine*, December 11, 493-499.

Fleming, H. Ed. (1990). *User Education in Academic Libraries*. London, UK.

Frisse, M. (1988). From text to hypertext. *Byte*, October, 247-253.

Gillespie, T. (1990). Computer currents: Hypermedia. *Library Journal*, October 1, 108.

Micco, M. (1991). The next generation of subject access systems: Hypermedia for improved access. *Information Today*, July/August .

Neuman, D. (1991). Designing library instruction for undergraduates: Combining instructional systems design and naturalistic inquiry. *College & Research Libraries*, March, 165-176.

Nielsen, J. (1990). *Hypertext and Hypermedia*. Boston, MA: Academic Press.

Penniman, W. D. & Dominick, W. D. (1980). Monitoring and evaluation of on-line information system usage. *Information Processing and Management*, 16, 17-35.

Rice, R. E. & Borgman, C. L. (1983) The use of computer-monitored data in information science and communication research. *Journal of American Society for Information Science*, 34(4), 247-256.

Schuerman, R. L. & Peck, K. L. (1991). Pull-down Menus, menu design, and usage patterns in computer-assisted instruction. *Journal of Computer-Based Instruction*, 18(3), 93-98.

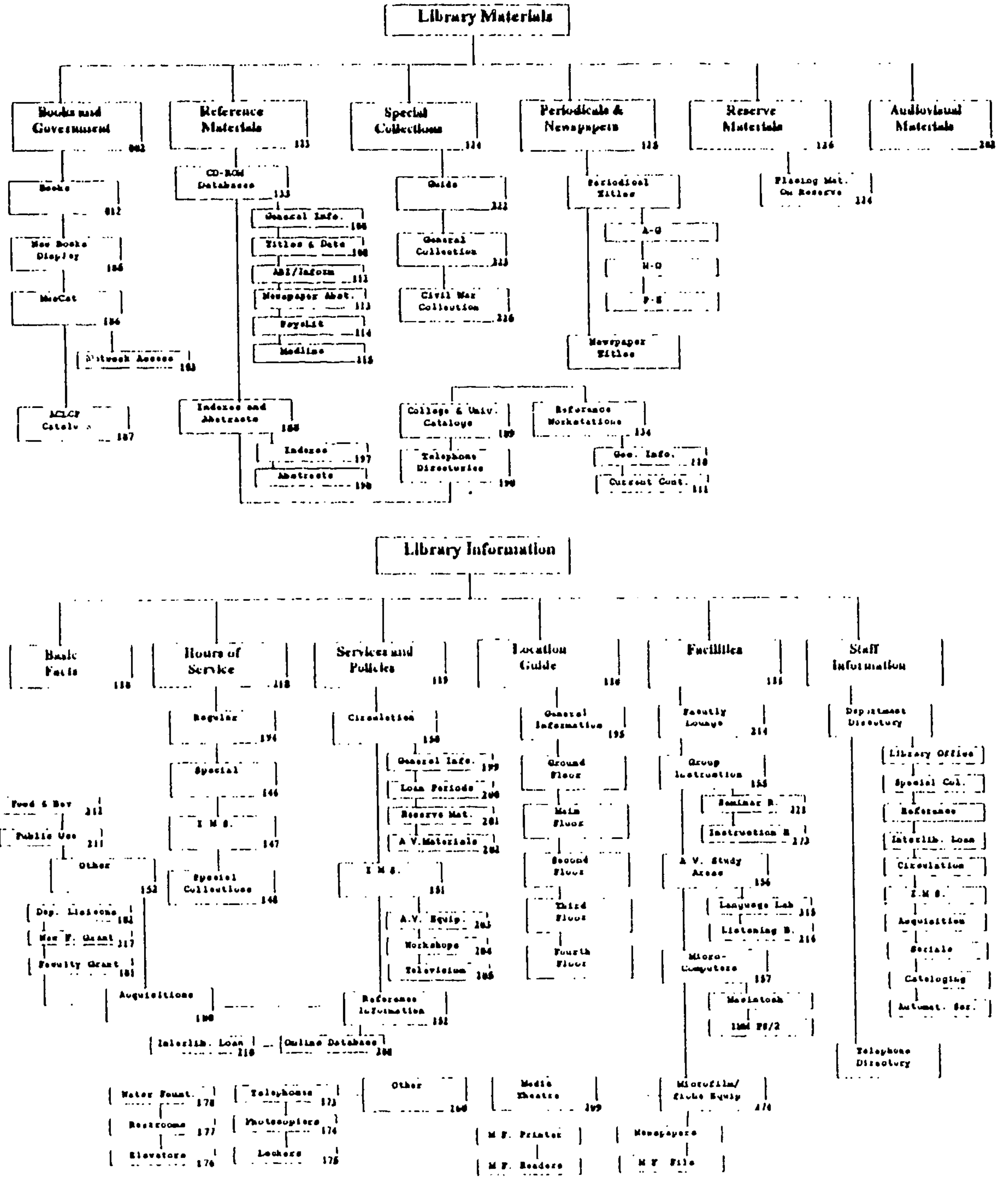
Waterworth, J. A. & Chignell, M. H. (1989). A manifesto for hypermedia usability research. *Hypermedia*, 1(3), 205-234.

Acknowledgments

I would like to thank Willis Hubbard, College Librarian, for his support and encouragement on LIMAS system. I also appreciate the comments and suggestions from librarians and faculty members at Gettysburg college. Finally, I am grateful to my student assistants for much of the excellent programming, screen design, and data input.

ERIC

Appendix. LIMAS Hierarchy Diagram



LIMAS TUTOR and the interactive multimedia CD-ROM software are not found within the LIMAS Hierarchy. They are both separate programs linked to LIMAS

Help, Main Menu, and **Index** buttons are found in all pages of LIMAS. "See Also" refers to linked pages from other parts of LIMAS (EX: See Also - Circulation Policies).

PANEL

Improved Classroom Interaction Through Multimedia

585

ERIC

Improved Classroom Interaction Through Multimedia

STEVEN L. EPSTEIN
Higher Education Consultant
IBM Corporation,
590 Madison Avenue, New York, NY 10022, USA
(212) 745-3835 SLEPSTE@NYCVMIC1.VNET.IBM.COM

In recent years, a large number of colleges have installed "master classrooms" in which computer and multimedia technologies are married with large screen projection to enhance the teaching process. This use of technology allows the professor to enliven the lecture by showing the computer image to all students. Depending on the subject and topic of instruction, this image can contain computer generated text, data or graphic images as well as full motion video that can be accessed from either a laser disc player, the local computer's hard drive or a multimedia network server.

Success with this mode of instruction has led early innovators to expand the installations of these classrooms on their campuses. As the price of the technology continues to decline there will be further impetus for expansion. The introduction of this means of using media in the classroom is as revolutionary today as the use of the blackboard was in the last century; however, it often does not change the way in which students participate during the lecture. That is, the professor still professes and the students still sit and take notes. Teacher style, class size and the sociology of the classroom do not encourage much two way communication during the "lecture."

Yet, the power of the computer and the power of the multimedia do not need to remain in the front of the room. Through the addition of small keypads at each student's seat, the class members can be enabled to interact with the professor and with each other. Thus, when the professor asks a question, all may respond. The aggregate answers can be projected and individual answers can be saved for later use by the professor. This technology allows the professor to improve attention, ensure comprehension, measure retention, and save time in the process of grading. Further, the combined use of the computer tools, the multimedia, and the interactive keypads allows the professor to cover more material in less time.

This session will employ multimedia to allow three professors to demonstrate the different ways in which technology is used on their campuses and to report on research as to the effects of the technology on student attitudes and learning and on the impact on faculty productivity. Material being developed by publishers for use in interactive multimedia classrooms will also be demonstrated.

BEST COPY AVAILABLE

Student Satisfaction with Keypad Response-Units: A Study Across Disciplines

MARLA J. FISCHER
Ancell School of Business
Department of Management Information Systems
Western Connecticut State University, Danbury, CT 06810, USA

During the past five years Western Connecticut State University and the IBM Corporation have participated in a joint research project aimed at evaluating an experimental instructional delivery system called the "Advanced Technology Classroom" (ATC). One feature of this innovative, electronic classroom is a student keypad unit that allows students to respond electronically to questions posed by the instructor. The design of this unit permits students to respond confidentially to instructor-initiated questions. Thus, although the instructor knows how each student is responding, other students are aware only of their own responses. The data collected via the student-response units are then pooled by a computer and displayed numerically or in bar-graph form on a screen. Students can see not only how the class is performing as a whole, but also how the class compares to previous classes responding to the same question. The keypad response units are not intended to eliminate normal student-instructor interaction, but to provide an additional form of interactivity that can be called upon during the course of a lesson.

One goal of the WestConn/IBM Project was to study the effectiveness of these keypad devices from the students' perspective. Specifically, we wondered whether students would prefer the keypad method to more traditional question-and-answer techniques used in the conventional classroom. In addition, we were interested in determining whether student satisfaction with the keypad units would vary as a function of academic discipline.

To answer these questions a study was conducted on students enrolled in three different content areas: education, management, and statistics. Each class was conducted in the ATC using the keypad units. The units were employed in a consistent way following established guidelines. At the conclusion of each course a survey was administered to evaluate student satisfaction with the keypad units.

The survey results indicated that students strongly preferred the keypad method over more conventional forms of responding. Moreover, these preferences remained in force across all subject areas with 83% to 93% of the students favoring the keypad units. Anecdotal statements from students also highlighted the keypads as an important component of the system.

Student Response System

Dennis R. Short
Technical Graphics Department
School of Technology
Purdue University, W. Lafayette, IN, 47907-1419, USA

A student response system (SRS) has been used experimentally in a classroom in the School of Technology for approximately two years. The system has been used primarily for summative evaluation and administrative tasks. Developmental work in the use of the SRS for formative evaluation and interactive instruction has also occurred. This work has been combined with efforts to implement electronic grading and distribution of evaluations on an automatic or semi-automatic basis.

The interest in the system has been primarily two fold: Does the system allow the instructor to improve the quality of instruction with no or minimum additional preparation costs and does the system allow the instructor to be more efficient in the administration of large or multiple-section courses?

Initial investigations in the use of the SRS were focused on its use as an instrument for administering short tests and quizzes. Concerns had been raised over the potential impact of synchronous testing (i.e., lock stepped test) on performance. Additional questions were raised concerning instrument effect and student attitude toward the devices. Data collected to this point indicates that no instrument effect exists for tests and quizzes up to 25 questions. Data for tests longer than 25 questions is being collected and evaluated. Students had expressed anxiety toward the system initially but this anxiety diminished significantly once experience in the use of the SRS had occurred.

Initial problems with the system were limited to instructor familiarity and insuring that display size and display quality were adequate. No equipment problems were found and the system has been quite robust in a daily student contact environment. The system is quasi-permanently wired in a 22 station classroom Some adjustments were required to accommodate visually challenged students

Formative evaluation and interactive uses have been focused on determining whether students are comprehending the material being presented and to elicit comments on the quality and effectiveness of the instructional presentation. What-if scenarios, Delphi-type evaluations and rank-ordering sessions have also been conducted.

Currently, plans and reques' for funding are underway for two additional installations. One is for a forum type classroom of approximately 75 seats and the other is for a large lecture hall of approximately 500 seats. The installation of the initial SRS was made possible by an IBM Study Grant.

A multimedia presentation for teaching video literacy

KAREN SWAN
Department of Educational Theory and Practice
SUNY Albany, Albany, NY 12222, USA

Video is the dominant media in American society today. It is the primary source of news and information for the majority of the American people. American children spend more time watching TV than they spend in school. It is essential that we teach them to take a critical attitude toward what they view via video. The development of such an attitude begins with an understanding of the "language" of the medium; an understanding of the techniques and conventions videographers use to make meaning from moving visual imagery. "Video Conventions" is an author-created, multimedia application designed to teach teachers the language of video so that they can, in turn, teach our children.

"Video Conventions" is written in *ToolBook*, a software package from *Asymetrix,* that supports classroom presentations which combine text, graphics, computer animations, sound, and video from a laserdisc on a single computer screen which can be projected to an entire class. "Video Conventions" integrates definitions of particular video conventions with video clips from the movie *Ghostbusters* which illustrate them. These are organized into sets -- lighting, sound, framing, camera angles, camera movements, and transitional devices. For each set, there is a computer animation which illustrates its distinguishing criteria.

For each set, there is also a student response page which utilizes the *Classroom Presentation Option (CPO)*, a hardware and software package available from *IBM*, to test learner understanding of the conventions in that set. The student response pages present learners with a series of video clips and ask them to identify the conventions they illustrate by choosing from a multiple choice list. Learners type their choice into an individual keypad. The *CPO* collects learner responses and displays them as a histogram, allowing the instructor to uncover and remediate misconceptions without embarrassing individuals.

"Video Conventions" has been a tremendous addition to my graduate classes in "Video and Education." Before I created the application, I had to give examples of conventions using videotape. I would discuss a set of conventions, then run a clip illustrating several of them and try to talk over it. Now, I can associate a definition with its illustration directly, show the illustration several times, stop motion on a particular frame, even step through the entire clip to make my points. The computer animations make it much easier to discuss defining criteria within a set.

Perhaps most importantly, the student response pages embedded in "Video Conventions" make it possible for me to test my students' understandings of video conventions and to provide feedback to them. Educators have long understood that the most effective learning situations provide opportunities for overt student responses (Kulhavy, 1977; Schimmel, 1983; Kulik & Kulik, 1988). My preliminary research (Swan, 1992) indicates that such findings apply to whole group as well as individualized situations. My students agree. "I will never look at video the same way again," they tell me, and that is exactly what I want to hear.

PANEL

Kids Making Media: A symposium about video and multimedia making practices of students in K-6 classrooms

ERIC

KIDS MAKING MEDIA:

A SYMPOSIUM ON VIDEO AND MULTIMEDIA-MAKING PRACTICES OF STUDENTS IN K-8 CLASSROOMS *OR* "MULTIMEDIA MEETS SOCIAL CONSTRUCTIVISM"

TED M. KAHN
Institute for Research on Learning (IRL)
2550 Hanover St., Palo Alto, CA 94304, USA
<Ted_Kahn@irl.com>

The use of video and multimedia is becoming more and more common in K-12 schools. However, in many cases, these media are being used primarily as a means to support more effective and enriched delivery of instructional materials and resources to individual students. While there are now many excellent examples of well-designed, interactive multimedia learning products, the use of these products in classroom settings tends to still be based around a model of learning as the transfer of content or skills into the heads of individual students. Such educational practice is often at odds with the modern learning sciences research which stresses an experiential, "hands on," and social constructivist approach to the nature of both school and everyday learning (e.g., Newman, *et al.*, 1989; Harel & Papert, 1991; IRL, 1992).

At the same time, there is a growing agreement about the importance of "media literacy" as a basic skill for all citizens, i.e., the ability to "read, analyze, evaluate, and produce communications in a variety of media—print, TV, computers, and the arts, etc. " (Kolkin & Tyner, 1991). A hands on approach to "multimedia making" is now possible for large numbers of students with relatively inexpensive video cameras and computers. Furthermore, when combined with the understanding that learning is fundamentally a social process, and that knowledge is socially constructed as a part of the ongoing participation of both students and adults in various "communities of practice" (IRL, 1993; Lave & Wenger, 1991), multimedia can open new avenues for student expression and participation in the practices of such communities

Participants in this symposium are all actively involved in developing students' multimedia literacy through providing them with direct experiences in making media. These three presentations may be viewed as mini-case studies, including: K-1 students learning to first make videos as the basis for multimedia work (Menkin & Raz), student teams designing and constructing "dream" environments of the future, using HyperCard and QuickTime (Mintz), to older students using specially designed multimedia composition tools to explore the creation of new types of interactive, multimedia documents (Soloway, *et al.*) In all these cases, technology is being used as a vehicle to help build, support, and sustain learning communities.

This presentation will discuss some of the cognitive and social design issues that are essential to the successful growth of media-making learning communities by first highlighting two successful, related projects with at risk students: MultiMedia Works (Pea, 1991) and the award-winning Rowland High School Animation Lab (Kahn & Master, 1992). Short segments of student work from each of these projects will be used to illustrate the power of student-made video and multimedia as a means to understand how these media can be a window into kids' social learning and creativity, as well as their worldviews.

ERIC

References

Harel, I. & Papert, S. (eds.) /MIT Media Lab (1991). *Constructionism.* Norwood, N.J.: Ablex Publishing Co.

IRL (1993). A new agenda for learning: "Putting people first." Palo Alto: Institute for Research on Learning.

IRL (1992). The school as a community of engaged learners. Palo Alto: Institute for Research on Learning.

Kahn, T. M. & Master, D. (1992) Multimedia literacy at Rowland: "A good story, well told." *T.H.E. Journal*, ***19***, 7, (Feb.)77-83.

Lave, J. & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation.* New York: Cambridge University Press.

Lloyd-Kolkin, D. & Tyner, K.R. (1991). *Media and You: An Elementary Media Literacy Curriculum.* Englewood Cliffs, NJ: Educational Technology Publications.

Newman, D.,P. Griffin, & M. Cole (1989). *The construction zone: Working for cognitive change in school.* New York: Cambridge University Press.

Pea, R.D. Learning through multimedia (1991). *IEEE Computer Graphics and Applications*, (special issue on multimedia), July, 58-66.

KIDS MAKING MEDIA:

"OVER IN THE MEADOW, WHERE THE STREAM RUNS BLUE, LIVE YOUNG KIDS WHO MAKE VIDEOS AND MULTIMEDIA, TOO"

WILLIAM MENKIN & SHULA RAZ
Yavneh Day School
14855 Oka Road, Los Gatos, CA 95030, USA
<Yavnehuw@aol.com>

Elementary teachers interested in integrating the use of video and multimedia into their own classrooms are often faced by a sea of questions: "What kind of equipment do I need? How many computers should I start with? What software should I buy? How will I ever learn how to use all of this? Where can I get money (grants) to support this?" While all these questions are relevant to developing a successful program, we hold that they must first be tied to more basic questions of pedagogical philosophy and classroom organization in order to be truly effective.

This presentation documents how we approached these issues in a combined K-1 classroom and how we began to integrate video and multimedia technology into our curriculum. Computers, video, and multimedia connections were used for instruction and enrichment as well as for creative tools for students to actively express themselves and convey their learning to others. The use of this technology supported cooperative learning in an integrated approach to art, math, language arts and science learning. It has also encouraged group creativity and personal expression.

Beginning with language arts and reading projects based on the well-known children's books, *Over in the Meadow* and *Frog and Toad are Friends*, we will present examples of how children in this age group used simple video equipment to enhance their understanding of literature, science, technology, and art. Their productions involved planning, scripting, group cooperation, and criticism. Through these experiences, children learned story sequencing and began to look at objects with the point of view of, "How will it look through the camera." Examples from other student productions will include videos of class science experiments, care and use of classroom materials, field trips, and classroom activities. Because the classroom lacked any video editing equipment, all video productions needed to be carefully preplanned and shot in chronological order. Where this may have made sophisticated video techniques impossible, it made a necessary virtue out of critical thinking skills involving planning, writing scripts, pre-production experimentation, and pre-shooting rehearsals.

In addition to video camera and VCR, the classroom was furnished with one Apple Macintosh LCII and three Macintosh Plus computers, CD- ROM drive, laser videodisc player and television. Students learned to manipulate Apple QuickTime™ movies from the CD-ROM drive, access Kid Pix and "QuickPix" files, and to place a group of pictures into a KidPix Slide show. Using commercial HyperCard™ stacks, they also learned to select images and pictures from laser videodiscs for use in sharing with other class members and writing activities.

We believe these activities promote student initiative and leadership skills. Students were encouraged to develop their own video ideas and work with others on tasks such as script reading and writing, brainstorming ideas, and putting their ideas into a shooting sequence. For younger children, these activities were a motivation to focus and increase their attention span. It provided many with just the pleasure of the aesthetics of producing in a new art form.

References

Kid Pix and *Kid Pix Companion*. Macintosh software. Brøderbund Software, Inc.. P.O. Box 6121, Novato, CA 94948. 800/521-6263.

ERIC

KIDS MAKING MEDIA:

READING AND WRITING IN THE 21ST CENTURY

ELLIOT SOLOWAY, MARK GUZDIAL, & KENNETH E..HAY
Highly-Interactive Computing Environments Project (HiCE)
Artificial Intelligence Laboratory, Dept. of EECS
University of Michigan
1101 Beal Ave., Ann Arbor, MI 48109, USA
<Soloway@eecs.umich.edu>

In education, the dominant mode of expression and communication has been based on words. Besides the clear educational value in learning to deal with the abstractions inherent in words, technology has made it easy to work with words. From styluses to the ball-point pens, from typewriters to the desktop publishing software, the central focus of technology has been on enabling everyone, not just a special elite, to manipulate words.

However, the mechanisms of learning, of self-expression, of communicating are not limited to words alone. Quite the contrary, in fact. Visual and auditory modes, for example, are powerful and used on a moment-by-moment basis in our everyday lives.

The ascendancy of computing & communications technologies in all forms -- main frame to hand-held, satellite communications to gigabyte networks -- offers education an unprecedented opportunity: no longer need there be any penalty for using the non-textual media in documents; no longer need there be any penalty for linking one's documents to those of our collaborators, our peers. Transmitting documents that incorporate sound or even video to a colleague half-way around the world will be the norm as the Nintendo Generation moves into middle and high school.

To explore that vision, we are developing and classroom testing software environments that enable students to read and write a new class of documents: multimedia, interactive documents. For example, MediaText (1992) is a multimedia document processor that makes it relatively easy to incorporate video clips, music clips, sound clips, animations, and still images, as well as words into one document. We refer to MediaText, with tongue in cheek, as the Brownie Instamatic of multimedia.

MediaText is in use in various classrooms in the Ann Arbor school district. For example, middle school science teachers use MediaText to create video-based instructional materials. In Community High School, MediaText has been used in composition courses as well as in science courses. Currently, an urban ecology class is using MediaText on a daily basis; youngsters routinely use multimedia materials as they write their observations in their journals. The major findings from all these experiences are these: teachers feel quite comfortable using MediaText to create materials, while students are able now to express themselves and communicate using the range of media supported by MediaText.

References

MediaText (grades 7-12) and *My MediaText Workshop* (K-6). Macintosh software. Distributed by Wings for Learning/Sunburst, 1600 Green Hills Rd., Scotts Valley, CA 95067, 800/321-7511 (developed by Constructive Instruments, Inc, Ann Arbor, MI).

DREAM CITY: MULTIMEDIA AND DESIGN IN THE APPLE VIVARIUM PROJECT'S OPEN SCHOOL

DAVID MINTZ
Vivarium Program
Apple Computer, Inc.
292 S. La Cienega Blvd., Suite #301
Beverly Hills, CA 90211, USA
<Mintz1@applelink.apple.com>

For the past six years, Apple Computer, through the Vivarium Program, has been involved in a long term collaborative research project with the Open School, a public, grade 1 - 6 magnet school in Los Angeles, California. Apple has introduced a variety of technologies to the school so that, by observing and learning from the teachers and students, we can develop richer, more functional media.

Currently the Open School has six double-sized classrooms, called clusters. In each of these post World War II bungalows two teachers work with 64 students. Despite the small size of the classrooms, there are 30 Macintosh computers per classroom, giving the teachers approximately a 2 to 1 student to computer ratio.

The Yellow cluster, with children in grades 3 and 4, has participated in and redefined a curriculum around building and designing a city, infusing technology into the project where appropriate. Over the years children have designed and built many versions of a city set 100 years from today. This year-long experience is woven into every aspect of the daily curriculum in the classroom. Various Macintosh tools, including HyperCard, Canvas, and MacWrite have been used to enhance the experience. Children have built HyperCard stacks that allow visitors to tour their unique parcel of the city as well as understand the whole city. Children construct the city, in their classroom, over a 30 by 40 foot foam core model. Although the city is actually built and HyperCard stacks can tell visitors about the city, it has not been until the availability of QuickTime that children could actually try to live in and experience life in their city.

This past year, children incorporated QuickTime movies into their HyperCard stacks, providing themselves and others with unique perspective of the city. Our presentation will discuss the limitations and potentials of HyperCard and QuickTime as extensions of children's understanding of their environment.

Trademark Note

QuickTime is a trademark and HyperCard and MacWrite are registered trademarks of Computer, Inc. Canvas is a registered trademark of Deneba Software, Inc.

PANEL

Techniques for Analysis and Evaluation of User Interactions with Hypermedia Systems

ERIC

Techniques for Analysis and Evaluation of User Interactions with Hypermedia Systems

WAYNE A. NELSON
Department of Educational Leadership
Southern Illinois University at Edwardsville, Edwardsville, IL 62026, USA

STEPHEN W. HARMON
Department of Instructional Technology
University of Houston-Clear Lake, Houston, TX 77058, USA

MICHAEL A. OREY
Department of Instructional Technology
University of Georgia, Athens, GA 30502, USA

DAVID B. PALUMBO
Department of Instructional Technology
University of Houston-Clear Lake, Houston, TX 77058, USA

With the emergence of hypermedia systems as a major platform for educational software comes the related problem of how to document and analyze learner interactions with such systems for the purposes of research and evaluation. There are a variety of interface design strategies that impact on how a system performs and should be evaluated. Many hypermedia systems to date have employed a "browsing" approach to interface design, but alternative designs are also emerging. Regardless of the type of interface, many questions can be generated when studying the interactions of users with hypermedia systems. For example, how many users chose to follow a particular link, and why did the learner choose one link over another? How does the choice of one link affect choices of subsequent links? What tasks are appropriate for guiding interaction with the system? What kinds of strategies do learners develop while working with hypermedia systems? These and many other questions need answers when designing, developing, and evaluating hypermedia applications for education.

A wealth of data regarding user behavior can be easily collected within many hypermedia development environments, including the sequence of navigation patterns, time spent at each node, use of help and orienting facilities, etc. The problem is that because of the nature of this data, traditional methods of analysis are not particularly effective. Researchers have had to develop new techniques for analyzing patterns of user interaction in their attempts to evaluate the design and effectiveness of hypermedia systems. There is a need to categorize and compare groups of users in order to determine the effectiveness of alternative system features, as well as describing characteristics of interaction by individual users within the same system.

This panel discussion focused on several techniques for collecting and analyzing user interaction data that have been found effective, including both quantitative and qualitative procedures. The methods range from simple frequency counts to complex path algebras, from protocol analysis to ethnographic studies. Representative techniques, including theoretical underpinnings, methods of data collection, and analysis procedures are described below. In addition, results from studies where the methods were employed are summarized, and recommendations for further research and expansion of the methods are suggested.

Characterizing User Interaction Using Path Algebras and Directed Graphs

Characterizing the interactions of individual users, or comparing groups of users can be accomplished using several methods derived from mathematical set and graph theories. Path algebras can be used to describe and compare the routes users take through hypermedia systems, and graph theory can be employed to construct network representations of user interactions. In the first method, user interactions are characterized in terms of various types of paths through a two-dimensional space. The moves from node to node of each individual are recorded in data files while the learner interacts with the system, and these files are then analyzed using algorithms that extract the nature of the "subpaths" of the total path. Types of "subpaths" can include rings, loops, paths, and spikes. The path that an individual learner follows during a session can be plotted as a function of time passed, giving an indication of the nature of the interaction with the system at different points in time. In addition, the frequencies of each type of subpath during the interaction can be used as a basis for comparison between individuals and groups.

This method was employed to characterize the interaction patterns of various learners who possessed different levels of prior knowledge for the domain of a hypermedia system. It was found that prior knowledge influenced interaction patterns (e. g., there were more "spikes" for individuals with lower prior knowledge). Extensions of this technique might utilize similar algorithms embedded within the hypermedia system to detect in real time some of the various interaction patterns, and to provide system interventions/guidance based on such analyses.

The second method employs graph theory to construct a network representation of the paths taken by individual users, and to compare networks generated by the Pathfinder algorithm. This method assumes that the more frequently a link was traversed by a learner, the more direct is the association in the resulting network that is constructed to represent the interaction pattern. Proximity matrices are constructed to represent the relative proximity of a node (in terms of the number of times it was accessed from a particular link) to every other node in the hypermedia document. The Pathfinder algorithm then extracts the representative network from the matrices. Various Pathfinder solutions from different data sets can be correlated to determine whether paths were similar between individuals or groups.

This method was also applied to the data mentioned above. Results suggested that prior knowledge may have influenced the interaction patterns of learners, as indicated by the patterns of correlations between the Pathfinder solutions of different groups of users. This method may also be extended by hypermedia authors to specify the initial organization of the nodes and links in a hypermedia document.

Using Social Interaction Methods to Analyze User/System Interactions

A quantitative approach to evaluating hypermedia systems has been adapted from social interaction research to assess feedback features in hypermedia-based intelligent tutoring systems (ITS) in order to determine the impact of one system's behavior on another's behavior. In social interaction research, it might be used to determine the impact of a spouse's behavior at time T on the behavior of the other spouse at times T+1, T+2, etc., where T is an observable behavior. The procedure has been used to assess ITS feedback in order to determine the learner's behavior at each decision point following feedback (either advice or error feedback). This same approach can be used for hypermedia analyses. In particular, it can help to determine whether a screen of information achieves its intended purpose.

For example, suppose a screen was created that depicts a process model of a computer, and that the model has links to various components of the system. These links are implemented with transparent buttons that are superimposed upon that component within the model. How likely is it that the learner will click on any one of these links (or a particular link)? The social interaction approach can give a statistical analysis of this scenario. The other important aspect of this technique is that it allows comparisons between systems. Two alternative screens for the process model described above might be designed with differing theoretical reasons for implementing the model in a particular way. Using the social interaction analysis technique, a researcher can test the competing theories in order to determine which is more effective for the intended learners.

Initial studies with this technique used dichotomous data (either positive or negative behaviors). Similarly, it is possible to categorize learner behaviors as either "the information was accessed" or "the information was not accessed". Then, a range of learner behaviors from one to perhaps ten behaviors following the decision point can be examined. A 2 X 2 contingency table can be generated for each click of the mouse. One dimension of this table would represent whether or not the learner accessed one of the nodes, while the other dimension would represent the learner behavior at other nodes. Using multiple contingency tables, comparisons of the learner's interactions could be made for each node that is linked to the original node. In addition, the interactions of multiple groups of learners (the two theoretically opposing depictions of the process model given above) could be assessed.

In studies employing this technique, chi-square values were calculated for each contingency table in order to determine whether learner decisions at various lag points varied as a result of alternative screen designs. It is also possible to calculate an F-score for comparisons between groups, allowing researchers to determine which is the more functional of the process models employed in the systems.

Using Qualitative Methodology to Generate Theory about User Interaction

The conceptually rich nature of hypermedia systems demands a methodology capable of dealing with this wealth. Imagine a situation in which a researcher sees one person wink at another one. Standard research methodology would dutifully note that subject one had winked one time. But the reason why this person winked would remain a mystery. Perhaps the person had something in his eye. Perhaps he was a spy conveying a secret message. Perhaps he was flirting. Whatever the reason, the why of this interaction is what is important to a qualitative researcher. The ability to describe the interaction from the subject's perspective is what has been called "thick" description.

Hypermedia systems are designed in part to allow users to actualize their existing semantic networks of information. If that actualization actually occurs, then obtaining the user's perspective becomes quite important. Any user's progress through the type of exploratory domains typical of hypermedia depends upon the user's perspective. Did a user go from one node to another because she was interested in that topic? Or perhaps because she was avoiding another node? Or perhaps because she was confused, or lost? Verbal reports have the benefits of allowing researchers to get directly at users' mental processes, providing massive amounts of data in a relatively short time. However, they are often difficult for users to generate, and can unduly influence user interaction with the system. Interviews have the benefits of not interfering with user interaction and of providing insight into user perspective. However, they are subject to additional levels of interpretation both by the researcher and the user. Path tracking has the benefit of providing an objective description of user behavior, but provides little or no insight into user perspective. The three methods of data analysis have similar advantages and drawbacks when used in research in user interaction in hypermedia.

Obtaining a thick description of how users interact with hypermedia systems allows researchers to generate theory about how to design these systems and how to evaluate learning (a difficult task where there are few, if any, specific learner outcomes). A combination of qualitative methodologies works well to enable researchers to examine user interaction in hypermedia. Three data gathering strategies (verbal reports, interviews and path tracking) and three data analysis procedures (constant-comparative analysis, data reduction, and discrepant case analysis) were used in a study to determine overall use patterns of hypermedia environments, and yielded interesting results that suggest a range of user behaviors, from seeking structure and confirming knowledge to exploring and "getting hooked" through curiosity.

Alternatives to "Browsing" Interfaces: Some Possibilities for Design and Evaluation

There is much discussion among proponents of hypermedia concerning the intrinsic advantages of non-linearity, associativity, and representing information via multiple media, but little is said about the interfaces of such systems. Some researchers presume a certain consistency between interface and information architecture when raising concerns of cognitive overhead and user disorientation in hypermedia systems. However, the way in which information is organized and stored does not necessarily correspond to the way in which that information is represented or interfaced.

It is interesting to consider the functional analogy between hypernetworks and the purported semantic network structure of long-term memory to illustrate this point. The way in which humans consciously recall and process information in long-term memory has little to do with the specific architecture of that information. The "interface" is effectively transparent in the sense that humans do not have to consciously engage in a "syntax" or set of procedures to access the information. Accessing an analogous network on computer generally requires an interface with substantially greater opacity. Hence, while the analogical validity between semantic networks and hypernetworks is probably valid, it is confounded by the fundamental differences in the way in which the information and user are interfaced.

In creating systems which purport to be intelligent, or better still, those which foster intelligence in humans through some form of directed adaptivity; it is imperative to clearly understand the ways in which we interface with the knowledge that we possess. Modeling semantic networks using hypermedia is worthwhile, but merely representing the model to users and allowing them to "browse" the information does not guarantee learning. Moreover, generalizing conclusions about the model or the underlying hypernetwork is necessarily limited to the specific interface elements in which the user engaged. The potential of hypermedia, in the abstract, may be significant, however two major limiting factors arise as such systems move from the conceptual to the actual. Both the interface and the representational structure of hypermedia are central to the development of more effective and efficient means of transferring knowledge. Much debate surrounding hypermedia, thus far, has centered on the underlying representational structure inherent in any hypermedia system and the degree to which it parallels human representational structures. An equally important consideration is the interface between the user and such a representational architecture.

There is a central distinction between the representational architecture of any knowledge-base and the ways in which users interface with it. While there is evidence of an analogous relationship between human long term storage mechanisms and hypernetworks, the variety of interfaces available to hypernetworks confounds the generalizability of this analogy. As development in this area continues, it is important to realize that both the knowledge representation and the interface issues are important as hypermedia-based learning systems are developed and evaluated. Non-linearity and associativity can be represented in numerous ways, thus allowing for a range of potential interfaces to meet the various needs of society as we move into the information age.

PANEL

Emergent Literacy in the Age of New Media

ERIC

Emergent Literacy in the Age of New Media

MONIQUE E. JUAN
School of Education
m.juan@um.cc.umich.edu

JEFFREY SPITULNIK
School of Education
spitulnik@um.cc.umich.edu

LESLIE MORRISON
Combined Program in Education and Psychology
l.morrison@um.cc.umich.edu

University of Michigan
610 East University Avenue, 1400E
Ann Arbor, MI 48109-1259 USA

Introduction

Young children love to participate in creative projects that enable them to communicate with others. As children begin to learn to read and write, they desire the opportunity to express themselves to their friends, teachers, and families using paper, pencils, crayons, paints, cut-outs, and so on. In our research at the Computers in Early Literacy (CIEL) Project at the University of Michigan School of Education, we have examined the developing nature of children's reading and writing as they use conventional tools (like paper, pencil, and storybooks) and computer tools (like *MacPaint*, *Story Writer*, and *Stories and More*). Our framework has been one of emergent literacy in which effective reading and writing develops over time as children explore the world of print and try things out on their own, with other children, and with adults.

Traditionally, being literate meant being able to read and write; however, today, many American children spend their out-of-school waking hours watching television and playing computer games, not reading and writing. In fact, many children frequently watch videos and use graphics-intensive computers in school. Children spend a lot of time in media-rich environments where the media includes more than just text. Furthermore, as computer-based interactive multimedia "storybooks" (and adult "books" for that matter) enter the mainstream, and as interactive television and other massive, interactive information sources become a reality for the general public, the concept of literacy is changing. Children are communicated to with many forms of media, therefore they must become literate in these forms.

This panel examines many questions raised by the introduction of computer-based multimedia into classrooms. How do we study the newer forms of mass media in the context of children's literacy? How can traditional theories of literacy be applied to the study of this new literacy? What is the developmental nature of children's conceptions about the computer? Will children effectively use interactive multimedia? Will they communicate with the tools we offer to them? What is the nature of this communication? Are the children able to use the tools? What are some design rules for the tools children will use to communicate using new forms of media? Do teachers use the computer to support the curriculum? What implications does this software suggest for future literacy programs and curricula?

Understanding the impact of new media on children, understanding the nature of the desires of children to communicate, understanding the teachers' desires to embrace new technology, and examining how children learn to use tools that facilitate a new age of communication will advance the understanding of software and content designers, teachers, communicators, and curricula designers.

Storybooks on the screen

MONIQUE E. JUAN

School literacy has evolved from slates and chalk to the use of various writing tools including computers and software. Emergent Literacy encompasses both reading and writing and the use of various tools to foster early forms of literacy. Because computers are a valuable tool when examining student's ability to write at different developmental levels, we can begin to determine if computer-based multimedia can enhance teachers' literacy programs by integrating technological tools into the classroom. IBM has developed *Stories and More*, a series of award winning trade books and non-traditional stories on the screen to expand early literacy programs and to determine if computer-based reading programs can be integrated into the language arts curriculum. In *Stories and More*, reading on the screen is presented interactively to students using various forms of media including sound, graphics and text.

Stories and More, a CD-ROM-based multimedia system, is used on a local area network. Features of the program include voice options that provide reading support for all 36 stories on the disk. (In the narration of the stories, the voice is digitized, not computer-generated speech, which can be adjusted to a moderate or slower reading pace.) Entire books are read aloud by the computer and students can choose specific words to be read aloud. Interactive computer activities with auditory feedback to provide immediate positive reinforcement introduce and follow up story themes and concepts for 16 core stories. Through these activities and suggested interdisciplinary classroom activities, students and teachers are challenged to incorporate story themes into their entire literacy program. An on-line library provides an additional 20 stories for children to explore without computer-based interactive activities. All stories are trade books that are available in print form to the students in the classrooms and can be repeatedly read by the teachers.

The primary focus of this research is to examine the behaviors of the students while they are engaged with *Stories and More*. What type of interaction do the students have with the computer and this specific multimedia software? Which type of activity do the students find most engaging and do those activities enhance students' comprehension at all levels? *Stories and More* offers three activities which focus on prior knowledge, comprehension, and applied knowledge. What implications does this software suggest for future literacy programs and curricula? What suggestions do the students have to improve the software? Early observations and research have begun to show students are highly engaged and several have shown some high order skills by discussing the books and actively persuading others to accept their view.

This research also observes how teachers introduce and incorporate *Stories and More* into their literacy programs as well as what teacher-controlled factors affect the utility of the computer. In the past, equity and utility, have been factors when determining if computers are regarded as literacy tools by both teachers and students. Do teachers use the computer to support the curriculum? Are the computer and activities which students engage in referred to and discussed? Do teachers value the computer as a valid literacy tool? If multimedia continues to be beneficial to children's understanding of literacy, how will literacy need to be defined and approached by teachers and curricula developers?

The site for this study is a public school with an Open School philosophy. Six teachers in multiaged classrooms agreed to take part in the evaluation of *Stories and More*. Teachers use the computer as a literacy tool, but not as the primary focus of their literacy program. Before the introduction of this literature-based software, students used *Children's Writing and Publishing Center* and *Primary Editor Plus* for more than one year. These are primarily word processing programs with clip art and drawing capabilities. Students have been using the computers as a writing tool, but with the integration of *Stories and More*, students may begin to regard computers as literacy tools that accommodate the various developmental levels present in multiaged classrooms.

Designing tools for children creating interactive multimedia

JEFF SPITULNIK

This research focuses on children's use of multimedia creation environments; the field of human-computer interaction is used as a basis for answering questions concerning the nature of literacy in young children and the use of tools to support literacy development.

Human-computer interaction research is concerned with improving the design of computer tools for ease of use. Computer-based literacy tools should be easy to use since students have the primary task of learning and creating; the tools should not put further burdens on the users – they should not distract or create obstacles to the goals of young writers. Tools should assist or at least enable users in accomplishing the primary tasks and they should have the *functionality* required by the targeted users.

The human-computer interface, or user interface, represents the part of the computer tool with which the user interacts. It is the communication channel where the computer represents its functionality and current state and where the user communicates his or her intentions and desires. When a user accesses the functionality of a computer tool and carries out the tasks for which the tool is designed, s/he is doing so through the user interface. The degree to which the user interface enables the user is a crucial factor in system design called *usability*.

Through a detailed analysis of children using multimedia creation environments, I explore how children use tools to communicate, including what features of the programs give children difficulty, what features are misunderstood, what features are easily remembered or confused, the nature of the communication, and so on. My goal is to provide a detailed description from which judgments about how tools of the future might be better designed to support children's emergent use of multimedia as the pervasive form of communication in their culture.

Using various cognitive and social frameworks, (for example, expert–novice comparisons, knowledge and procedure representations, misconceptions, issues in motivation such as self-efficacy and value systems, various interactions within classrooms, and so on), I will examine the design of several computer-based, interactive multimedia creation environments for young children.

First grade children in a low income, elementary school in an urban setting use networked Macintosh LC computers with *KidPix*, *The Playroom*, and *Dragon Tales* (multimedia storybooks, encyclopedias, activities, and tools for creating multimedia stories).

Children will work alone and in groups to produce presentations, stories, and portfolio-like notebooks about subjects of their choice. In similar creative contexts during which children use traditional literacy tools (pencil, eraser, and paper), first graders usually write about books they have read, families, playing with friends, recent vacations or field trips, and so on.

I will primarily examine the usability of the tools, but I will also be discussing the subject matter chosen by the children, how the children choose to represent the material with new multimedia tools, and the general characteristics of how the children use the tools. I will be looking at both the cognitive and social dimensions of the use of computer tools in an educational setting. This study has three main phases: (1) A heuristic analysis phase during which I analyze the computer-based tools for usability and utility factors; (2) A classroom observation phase during which I study the use of the tools in context; (3) A prescription phase during which I suggest design principles for computer-based interactive multimedia creation environments for children.

Using Theories of Emergent Literacy to Examine Literacy in the Use of New Media

LESLIE MORRISON

Emergent writing is part of a broader domain of emergent literacy (Sulzby, 1990). Forms of emergent writing include scribble, drawing-as-writing, letterstrings, invented spelling, and conventional orthography. These emergent forms of writing display parts of the concepts that children infer about writing; thus, emergent writing is conceptual and not just imitations of adult behaviors. However, children's writing does not occur in one invariant, hierarchical order, in which children move from one underlying conceptualization to another. Instead, children's literacy development seems to move across various forms of writing, especially when writing different kinds of texts and using different types of media (Sulzby, 1990).

With computers entering the mainstream and the classroom, we must expand our concept of literacy to include the computer. Furthermore, with computer-based interactive media, we must examine what kind of literacy or literacies children are experiencing as they enter the 21st century (Sulzby, in press). This case study examines a first-grader's writing on and off the computer across several sessions during the school-year. In the study, we addressed several questions, including: (1) How does the computer allow or constrain the use of emergent forms of writing that children use off the computer, and (2) What is the developmental nature of children's conceptions about the computer?

In the CIEL project, we have found that kindergarten and first grade children from both middle income and low income settings can use the computer successfully as a writing tool (Sulzby, 1992; Sulzby, Olson, & Johnson, 1989; Spitulnik & Sulzby, 1992). If the computer enables the child to use all of the emergent forms, they will use emergent forms. However, we have found that scribble has been continually under-represented in our distributions. We have also found that children's composition behaviors seem to parallel those off the computer. In addition, children show such compositional behaviors in computer writing as planning, revision, and rereading. In this study, we have also found evidence of conceptual conflict in children's writing on the computer. Since children's transition from scribble, drawing, letterstrings, and inventive spellings into conventional writing is gradual, children often reach points in their development when they wrestle with their changing concepts. Examining conceptual conflict may help further develop our understanding of children's thinking as they make the transition into conventional literacy (Sulzby, in press).

This study has many practical applications for multimedia use in the classroom. For example, it demonstrates the computer as an important tool for literacy learning in the classroom and shows how children's conceptions of the computer develop through knowledge and use of the computer in the classroom. In addition, it may help software designers better understand children's literacy development when using the computer as a writing tool. Furthermore, it helps parents, teachers, and researchers expand and define our changing concept of literacy.

References

Spitulnik, J., & Sulzby, E. (1992, December). Issues in human-computer interface: What young children show us. Paper presented at the symposium, Computers and the Transition from Emergent to Conventional Literacy at the National Reading Conference, San Antonio.

Sulzby, E. (1990). Assessment of emergent writing and children's language while writing. In L. M. Morrow & J. K. Smith (Eds.), *Assessment for instruction in early literacy*. Englewood Cliffs, NJ: Prentice-Hall.

ERIC

Sulzby, E. (in press). I thinked in my mind: Keeping young children's powerful thinking alive. In J. N. Mangieri & C. Collins-Block (Eds.), *Advanced Educational Psychology: Creating effective schools and powerful thinkers* (tentative title). Fort Worth, TX: Harcourt Brace Jovanovich.

Sulzby, E. Oslon, K. A., & Johnston, J. (1989). The computer and young children: An emergent literacy perspective. (Working paper No. 1, Computers in Early Literacy (CIEL) Research Project). Ann Arbor, MI: Institute for Social Research.

Acknowledgments

Support for this research has been from University of Michigan, Spencer Foundation, Office of Educational Research and Improvement, International Business Machines, Incorporated, Jostens Learning Corporation, and Apple Computer, Incorporated. Special thanks to Dr. Elizabeth Sulzby, principal investigator, Computers in Early Literacy Project (CIEL), University of Michigan.

Short Papers

This Computer Is Great, But Will It Mow My Yard?

LARRY S. ANDERSON, Ed.D.
Director, Instructional Technology
College of Education
Mississippi State University—Mississippi State, MS 39762, USA

HyperCard and LinkWay are the two premier tools that have leveraged teachers into a position where they use computers rather than just learn about them. This phenomenon can be explained with a simple analogy familiar to most people.

Picture it. A young man enters his garage on a warm Saturday afternoon and removes a lawn mower. He reaches down, gives the rope starter a tug, and begins pushing the implement back and forth across his lawn. Little by little, he strives to accomplish the main goal in his mind—to improve the appearance of his property. Eventually, the task is completed, so he cuts off the lawn mower, puts it back into the garage, and returns to his house for a cool drink of water.

In a few years, the young man's son has reached the age where he, too, can mow the yard. Consider the instructional process undergone by the man as he teaches his son. He shows the lad how to remove the lawn mower safely, how to refill the mower with gasoline, how to navigate the trees and bushes, how to avoid the rocks and ditches, how to make sure that all the grass gets cut, how to cut the mower off, and how to return it to the garage safely. Mission accomplished!

Some important questions are raised in an analogy between this lad's learning to mow the yard and a teacher's learning to use a computer as an instructional aid. Did the man or boy discuss the physics involved with friction induced by the mower wheels as they passed over the ground? Did the man disassemble the mower's engine so the child could see and learn the names of all the internal parts? Did the father teach the child all about the peripheral devices that can be connected to a mower, such as leaf bags, thatching blades, etc.? Did the father make the son take a mowing literacy test until he achieved a passing score? Did the child have to learn names of a variety of vendors who sell lawn mowers and mower parts? Did the father prevent his son from operating the mower until the lad could describe the functioning of an internal combustion engine? Was the son required to demonstrate knowledge of all the various types and configurations of lawn mowers (manual push, electric push, gas-operated push, riding, etc.) that are encountered typically in the lawn mowing business? What, exactly, was necessary in order for the child to achieve the same goal the father had in mind when he mowed the yard initially? Further, what could the father do to entice the child to look forward to mowing a yard? Would the father find it desirable that the child enjoyed his experience of mowing and could see the benefit of this activity to his life in the future?

The same kinds of thoughts and activities can be applied to educators' use of computers in the classroom. While it is not disputed that the well-educated computer user might be able to work out of a difficult situation, such knowledge is not essential to find worthwhile, pragmatic application of computing to instruction. Many, but certainly not all, teachers approach the infusion of computing into their instructional programs with fear and trepidation. What can be done to prevent this negative feeling? Moreover, what can be done to ensure such a positive experience that the teacher will look forward to using computers?

Hypermedia is the natural environment in which teachers discover an entirely new realm of excitement for learning. No longer do they have to be suppressed by the burden of having to learn to write long codified programs in BASIC, Pascal, or Logo in order to create meaningful lessons that target their individual students. Most people respond positively to a visual stimulus; therefore, the possibility of creating something in a visual environment elicits such positive response among teachers that they find sustained, intrinsic benefit from using hypermedia.

During the summers of 1991 and 1992, faculty members in the Department of Technology and Education at Mississippi State University (MSU) offered a special topics course, Introduction to Hypermedia, to Mississippi teachers who were enrolled in a program that led to a computer education certification endorsement. Historically, teachers had been expected to complete courses in which they learned all the traditional information about computers—bits and bytes, ROM and RAM, serial and parallel, input and output, immediate and deferred execution, GOSUBs, GOTOs, and WAIT WHILEs. Far too many teachers were seen leaving the traditionally taught courses in tears and declaring, "I'll never touch another computer as long as I live!" Faculty determined that there had to be a better way.

A technology-rich classroom situation was created for the hypermedia course in which the positive experiences these "average" educators encountered were allowed to flourish. Teachers created fantastic learning modules using HyperCard and LinkWay—and became increasingly excited throughout the process. These teachers were bathed in a non-threatening, success-oriented environment having sufficient support such that they were not allowed to fail. As they enjoyed success on their initial projects, they felt the confidence necessary to continue building instructional modules. Teaching faculty noticed that teachers did not lose their focus on the subject matter that was the object of their modules. Rather, they were so involved in creating interactive lessons for their students "back at school" that they exclaimed often to one another how they had encountered new ways to present material which had become stale over time.

Hypermedia had served as the catalytic vehicle in which teachers were carried through a process of instructional materials development and they did not <u>have</u> to know all about bits, bytes, interfacing, networking, BASIC programming, and the other myriad of "essential" topics included in most computer literacy courses. Now that teachers had seen how easy it was for them to use a computer, they felt comfortable to go back and learn about how many bytes of memory would be consumed by a particular digitized sound.

Instruction in the operation of necessary peripheral devices was kept at such a low keel that teachers did not feel intimidated by the technology. Regularly, they were told, "Using this videodisc is no big deal," and "digitizing audio is no big deal." Teachers were not protected from the nomenclature associated with videodisc players and the cables necessary for them to operate. Rather, the terms were incorporated as a natural, almost seamless, part of the conversations. The desire was to make the concept of digitizing audio seem just like creating a word processing file. This got them started. Later, when they had overcome any initial fear, we showed them the more intricate, detailed information about video and audio digitization. Too, "hooking it all up" was simplified as much as possible at first. The goal here was to make the teacher believe, "I really <u>can</u> do this stuff."

The emphasis on infusion of various technologies prior to formal instruction on the technical characteristics of each device allowed the teachers to concentrate on the instructional process. This arrangement helped the teachers overcome electronic minutia and gadgetry. Sometimes, teachers were encouraged to incorporate a videodisc or CD-ROM clip into their interactive hypermedia lessons without being told about the particular peripheral device. We attempted to make them see the value of using an instructional <u>resource</u> for the sake of its benefit, not for the sake of saying they had used interactive videodisc! Only after they had integrated their new resource successfully in the lesson did we point out how advantaged their efforts were because they had used this neat, new technology called interactive laser videodisc. A typical response from these students/teachers was, "Wow! I really did that!" This positive inspiration was evidenced in the quality of their overall instructional plan. An added bonus was that they could hardly wait to get back home and expand their idea into something their students could use repeatedly.

Hypermedia, as a visual medium, enhances the opportunities for users to profit from watching others work on similar projects. Graphics-oriented software entices the user/developer to explore others' projects for the purpose of discovering how someone else approached a particular problem (or the solution to some problem). Because each potential developer comes to hypermedia with an individual agenda intended to target a specific need, perusing work created by someone else often gives birth to a parallel idea borne by the new developer. For example, perhaps a second-grade teacher created a vocabulary lesson using hypermedia and his/her work was available for downloading from a bulletin board system. Suppose a high school Spanish teacher logs onto the bulletin board system and downloads a lesson. The foreign language teacher browses the project and is impressed with the idea that this same procedure could be used to great advantage in teaching Spanish vocabulary words to his/her students. Probably, the elementary teacher had no idea that his/her work would be found beneficial to other teachers, especially those in various disci lines. Due to the graphical nature of the project, the foreign language teacher could see direct implementation with a minimum of revision.

As teachers create hypermedia instructional lessons, they find that revision of their work requires minimal labor, especially as compared to the same project created by traditional means. Before the lesson is distributed to the masses of students, the teacher normally will want a sample set of students to run a pilot test on the lesson(s). Once again, in this process, the beginning hypermedia-developing teacher recognizes positive aspects of creating in hypermedia. If students have a problem with some particular portion or they recommend strengthening the project in a specific area, the teacher can modify/enhance the work with little effort, comparatively. Had this work been done in a traditional, line-oriented programming language, many teachers would not be willing to make any changes. As a matter of fact, the program probably would have been discarded since revision is so painful with line-oriented programming. Again, hypermedia is the key to the construction and revision of meaningful instructional materials.

Efforts by MSU faculty to introduce teachers with widely diverse backgrounds and interests to hypermedia have been rewarded by a near-constant environment of success. Teachers have called and written to report their success stories as they used their hypermedia projects. They have requested more training and opportunities for exposure to various forms of hypermedia. Remember that most teachers prior to this experience were going back to their schools declaring never to touch a computer again! Many teachers have bought their own personal computer as a result of their discovering this method whereby they can be successful in instructional materials development, through which they can reach children much more quickly and with less effort, and with which they can share learning experiences with their peers. Substituting hypermedia for traditional approaches to teaching teachers to incorporate computers into classroom activities has been one of the best efforts attempted in the College of Education at Mississippi State University.

Suppose one were to face the group of teachers who participated in these introductory hypermedia courses and ask them to answer the question posed in the title of this treatise. The answer would come back resoundingly, "Yes, this computer is great and it CAN mow my yard!"

A CBI/CMC Prototype for Legal Accounting Education at a Distance

DAVID ANNAND
CRAIG CUNNINGHAM
KONRAD MICHALSKI
Faculty of Administrative Studies
Athabasca University, Athabasca, Alberta, Canada T0G 2R0

This distance-education prototype features elements of print, computer-based instruction (CBI) and computer-mediated communication (CMC). It is being tested in part of a non-credit introductory accounting course offered to Alberta students-at-law. Currently, six out seven of the lessons within the course are delivered to most students via an on-line connection to Athabasca University's AT&T 3B2 minicomputer. The present on-line delivery system has several instructional and administrative deficiencies. The material is strictly text-based and very linear; opportunities for interaction and feedback are limited, learner control is minimal and communication capabilities are limited to essentially an electronic "bulletin board" format, with no means for student-student or one-to-one instructor-student interaction. The major instructional design goal of the project is to minimize these deficiencies while maintaining one of the major advantages of the course - independance of learner and instructor in space and time.

The prototype uses print as the primary instructional medium because of the nature of the lesson material (rule presentation and explanation), lower production cost and time, and print's ubiquitousness. However, the unique instructional capabilities of CBI and CMC are used to complement the print-based material. The Macintosh-based CBI section (developed using Authorware Professional) has a graphical user interface featuring both click/touch and pull-down menus, a "Comments" facility to allow students, for example, to identify problem areas in the CBI material, and various learning aids and navigation features. It also contains a seventy-item question bank. When selected by the student, a subset of these items is presented based on instructor-specified parameters. Student progress is monitored and displayed at all times. Tailored remedial feedback is provided based on learner response. The CBI component also contains an interactive simulation of the accounting process, featuring graphics, colour and animation. Twenty examples of trust and general account transactions are presented. For each example presented, the learner is required to select the appropriate (displayed) general ledger accounts. An animated sequence then illustrates how this information "flows" through the accounting system to the income statement and/or balance sheet of a hypothetical law firm. The financial statements show "before and after" versions to highlight the effects of each transaction. The scenario presentation, journal entry recording, general ledger account posting and financial statement summary is done on one screen. The simulation also provides detailed feedback to the learner as correct or incorrect choices are made.

The CMC component is invoked from anywhere within the CBI lesson. It uses a commercial electronic mail system to provide cost-effective (i.e., off-line) and direct student-to-student communication (one-to-one, one-to-many) at a distance. In addition, student performance data collected by the CBI review and feedback section and any messages filed using the "Comments" feature are automatically transmitted to a remote mail server whenever e-mail communication is initiated by the learner.

Review of the Literature as Related to the Pedagogical and Instructional Issues of Hypermedia.

PATRICIA RYABY BACKER & JOSEPH K. YABU
Division of Technology, San Jose State University, San Jose, CA, 95192-0061, USA

Because of the recency of Hypercard and hypertext, most research has focused on developing courseware and applications of hypertext for instructional purposes. Although there have been many different applications of hypertext as instructional vehicles, little work has been done in the evaluation of this media as an instructional tool or on its efficacy with differing school populations. According to a few researchers, Hypercard has some significant differences when compared to traditional Computer Assisted Instruction (CAI). Jonassen (1986) cited a benefit of hypertext in that it breaks traditional sequential processing and permits text presentation to be locally modified by the user to meet individual needs. Stanton and Stammers (1989) stated that the advent of hypertext allowed the design of courseware without any predetermined structure that made the designer's task easier and allowed greater flexibility for the learner to structure the learning environment to suit their own learning needs. In this way, hypertext supports nonlinear access to information and learning. However, in a study comparing performance differences in structured and unstructured environments, they found that the ability of students to function in an unstructured learning environment depends upon their individual learning styles. Campagnoni and Ehrlich (1989) found that individuals with better spatial visualization skills were faster in retrieving information in a hypertext-based help system than those with poorer visualization skills. In addition, this study highlighted another important feature of hypertext: hypertext seems to cause new strategie .n learning by browsing rather than by employing indices or predetermined menus.

Related to the efficacy of hypermedia instruction, Higgins and Boone (1990) conducted two studies to field-test hypertext computer study guides with forty ninth grade students (ten with learning disabilities, fifteen remedial, and fifteen regular education). Their findings indicated that the hypertext treatment was as effective as the lecture method with posttest scores higher for the computer study group. Backer (reported in Backer & Yabu, in press) compared the effectiveness of hypermedia instruction versus traditional methods using a self-paced, hypertext tutorial on introductory concepts of manufacturing and engineering drawing. She found that students who used the hyp[illegible] instruction had significantly higher post-test scores than the control group.

Refere[illegible]

Backer, P.[illegible] & Yabu, J.K. (in press). Hypercard Tutorial as an Instructional Strategy: A Comparative Study. *Journal of Industrial Technology.*

Campagnoni, F.R., & Ehrlich, K. (1989). Information retrieval using a hypertext-based help system. *ACM Transactions of Office Information Systems*, 7(3), 271-291.

Higgins, K. & Boone, R. (1990). Hypertext computer study guides and the social studies achievement of students with learning disabilities, remedial students, and regular education students. *Journal of Learning Disabilities*, 23(9), 529-540.

Jonassen, D.H. (1986). Hypertext principles for text and courseware design. *Educational Psychologist*, 21(4), 269-292.

Stanton, N.A., & Stammers, R.B. (1990). Comparison of structured and unstructured navigation through a CBT package. *Computers and Education*, 15(1-3), 159-163.

The Play's the Thing: Students as Theory Builders in the Technologized Drama Classroom

CLAUDIA BARNETT and ERIC D. WALBORN
Computers in Composition and Literature Program, Department of English
The Ohio State University, Columbus, OH 43210-1370, USA

What is a text? To whom does a text belong? How does one determine the meaning of a text? Our project takes these questions about textuality, authorship, and meaning central to poststructuralist literary theory and, utilizing computer-supported multimedia, applies them to the study of drama. Drama is a vehicle of performance, its text merely an outline, a set of instructions awaiting "completion" through production or imagination. Such completion may be achieved in an exciting new way through the multimedia modules that we conceive. In order to be used by students as tools to investigate and create theories of textuality, meaning, and authorship, however, the multimedia modules must reflect the nature of student inquiry while responding to the evanescent nature of performance and interpretation that defines and identifies drama.

Reading requires audience participation, as Umberto Eco reminds us: "Texts are lazy machineries that ask some one to do part of their job" (*The Role of the Reader* 214). Readers must make connections between events, themes, and clues, connections which Eco has termed "ghost chapters." Essentially, active readers become writers as they compose such ghost chapters in their minds. Reading dramatic scripts provides the most obvious example of the necessity of a reader's "writing" the white spaces in the text: in a printed play script, the ghost chapters are intentionally left blank (the intonation, the setting, the props) to be filled-in by actors, directors, and set designers. Such texts, however, are more often read than seen, and all the work is left to the reader. Multimedia can facilitate ghost-writing and help make reading drama a more visual experience.

A Movie Star Has to Star in Black and White (1976), by Adrienne Kennedy, is a mysterious, provocative, yet difficult to read play: the set is abstract; the language is repetitive and oblique; the time-frame is non-linear. "Marlon Brando" and "Bette Davis" coexist on stage with Kennedy's creations—Clara and her family—and "Davis," "Jean Peters," and "Shelley Winters" recite Clara's lines for her. This play, set against the backdrop of scenes from *Now, Voyager, Viva Zapata!,* and *A Place in the Sun,* serves as the prototype for our multimedia development project. Using multimedia components, students will be able to access these movies and study their relevance to the play. Other modules will enable them to design their own productions. They will make directorial choices regarding space, time, delivery, and tone. They will tell Kennedy's "movie stars" where to stand or how to move and see how these decisions create and alter the relationships among the characters in the play. Hypermedia components, including criticism and reviews of past performances, will similarly inform their productions.

During Autumn quarter, 1993, the *Movie Star* project will be presented as a rough shell of texts, stills, and sounds, to an introductory drama class whose task will be to complete that shell with its own ghost writing and to flesh out the interpretation for future students. These students will be actors, directors, dramaturgs, and stage managers, and then they will watch their own performances and become their own audiences; they will play all roles, viewing the "final" production from a multiplicity of perspectives, complicating and enriching their theories of drama.

The Ethics and Aesthetics of Rich Media Computer Human Interfaces

PAUL BROWN
Department of Art & MSU/NSF Engineering Research Center
Mississippi State University MS 39762 USA

Now that we have developed and are introducing rich media computer human interfaces it is important that we develop a taxonomy that can help us appreciate the power and value of these systems, quantify them and assess their creative and ethical impact. To date the application of semiosis has been to natural languages and human-human communications. Elsewhere (Brown '93 a,b) I have suggested that the Semiology of Charles Sanders Peirce can be applied to artificial languages and in particular to their use in human computer interactions. This analysis offers insight into the aesthetic (creative) and ethical implications of computer human interface (CHI) technology and the way it is being developed and used.

Two potentials that seem to essentially identify the uniqueness of the digital computational system are: communication between two, or more, humans that is mediated and/or enhanced by an "artificial intelligence" (AI) and; communication between a human, or group of humans and an AI where the term AI is used extremely loosely. These potentials also emphasize the importance of the computer human interface.

One of the most successful developments in interface design has been the use of metaphors. The most famous is based on Alan Kay's Xerox PARC project, Smalltalk, and popularized by the Apple Macintosh Desktop metaphor. Although this kind of interface goes a long way towards reducing user intimidation it nevertheless reinforces the misapprehension of the computer as a productivity/automation device. Also, because of its strict definition and use of external referents (like "desktops"), it limits human flexibility, creativity and exploration. My recent research attempts to identify the causes of this undesirable dichotomy where sophisticated CHI's seem to offer access only by forsaking a more comprehensive appreciation.

It is important that we find ways of developing the computer human interface so that it: can reduce user intimidation and encourage awareness and usage; does not disguise the inherent nature and potential of computational systems (conforms to an ethical paradigm) and; enhances (encourages) the user to develop novel and unique solutions that are not preprogrammed into the software or interface and endow an aesthetic appreciation and usage.

The Semiology of philosopher Charles Saunders Peirce enables a semiotic analysis of CHI: Symbolic Interaction includes alphanumeric command languages and command line interpreters like DOS, JCL, Etc.; Iconic Interaction, the current paradigm, includes the Smalltalk derivatives like Macintosh and Windows; Indexical Interaction includes immersion interfaces (virtual reality) and some rich media interfaces (multimedia input and output). The established associations and attributes of these terms enable us to better comprehend the limitations and potentials of different CHI techniques and so be able to make better informed, creative and efficient design and authoring decisions.

References

Brown, P. (1993a), Symbiosis, Semiosis and Communication Technology, *Proceedings of Inner Space/Outer Space 93*. Southern Humanities Council. To appear.

Brown, P. (1993b), The Ethics and Aesthetics of the Image Interface, *Proceedings ASIS 93 Mid-Year Meeting*, The American Society for Information Science, Silver Spring MD. To appear.

Hypermedia and Performance Support

CHRISTOPHER DEAN, COLIN WATT, QUENTIN WHITLOCK & SARAH WILKINSON
Dean Associates, 9 The South West Centre, Archer Road, Sheffield, S8 0JR, England

The rapid rate of change in business practices due to competition, changes in the world economics, technological developments and so on is challenging businesses of all sizes to react speedily and effectively to remain competitive. The development of computer technology - both software and hardware - has reached the stage where it is possible to think of designing and developing systems to support members of a company so that they can do their work effectively with current information at their fingertips and without constantly having to ask others how to carry out functions with which they are not totally familiar.

The concept of Electronic Performance Support Systems (EPSSs) has developed over recent years to describe computer systems that provide users with information, training, expert guidance and other forms of support while they are performing their job. The job itself is not necessarily computer based. The concept of an EPSS is a natural development from, and bringing together of, a number of application areas that are in regular use on many computer systems - areas such as databases, computer based training, electronic books, expert systems and help facilities.

Most EPSSs to date have been developed as one-offs to address problems in specific organisations. The research that has been carried out by Dean Associates has looked at the fundamental design methodology for EPSSs and is attempting to define a generic method of specifying a performance problem so that a solution can be developed efficiently to support any business where performance is an issue. This ambitious target can only be achieved by using tools that are already available as the building blocks. The keystone of the strategy is the adoption of an object-oriented approach to the design. Fitting in well with this is the adoption of Information Mapping techniques for the presentation of information. It will be evident that a typical EPSS will rapidly become useless if the information held in it cannot be readily updated. Object-orientation and Information Mapping allow a reliable and practical updating strategy to be adopted.

Two useability factors should be integral to the design of any EPSS. These are an effective user interface, preferably, but not necessarily, a graphical user interface (GUI) and a friendly and flexible method of navigating through the system. An EPSS of any size is going to have a very complex structure and will cater for different levels of staff, conceivably from senior executives to workers on the production line and sales order clerks. The use of hypermedia techniques to allow users to access the information in the EPSS in a flexible fashion is important but raises problems when updating becomes necessary. Part of the research has concentrated on developing a strategy that allows many of the benefits of hypertext to be adopted without the inherent difficulties of updating that the sophisticated use of hyperlinks causes. The strategy will be described and discussed in this short paper.

The Effects of Analogies: What Matters Most?

JERRY P. GALLOWAY
Indiana University Northwest
Gary, IN, 46408, USA

Comparative language includes a variety of different types of speech. Metaphors, for example, through word or phrase substitution suggest a comparison, although indirectly. The metaphoric figure of speech implies that a likeness exists even though the comparison is not explicitly stated. An analogy, on the other hand, asserts a particular comparison directly. Necessary attributes and processes are targeted for comparison while incidental characteristics are deemed insignificant and of lesser importance. Analogies can help build more accurate or more complete mental models (Gentner, 1983; Gick & Holyoak, 1980). An analogy which perfectly represents comparison relationships can be extremely useful in visualizing, internalizing and developing an understanding of a target concept.

Galloway (1992, a.) has provided evidence that teaching educational computing with analogies can improve the conceptual development of what students learn and the quality of explanations of computing. Obviously not all analogies are equally representative in their comparisons and some may initially seem quite poor. However, while it seems obvious that a good analogy would be preferred over poor, the most important concern, to facilitate conceptual development for students' understanding, does not require the best possible analogies.

Two different groups of beginning computing students were taught with computing analogies (Galloway, 1992, b.) which were divided (and used separately) into the best and worst based on an independent evaluation by three experts. In spite of the divergent treatments, results indicated absolutely no difference between the two groups on a pre/post written test, essay responses kept in a personal journal as well as the course final exam.

This study supports the view that, through proper development and student involvement in analyzing both similarities and differences of analogies, both good and bad analogies can apparently be equally effective in helping students improve understandings. There may be no such thing as a good or bad analogy. Such labeling may be inappropriate as analogies are selected for use in the classroom. The learning process lies in analyzing relationships and attributes in the target and comparison domains. Analogies are useful so long as they facilitate that process.

Galloway, J. P. (1992, a.) Teaching educational computing with analogies: A strategy to enhance concept development. *Journal of Research on Computers in Education, 24*(4), 499-512.

Galloway, J. P. (1992, b.). *Analogies in educational computing.* East Rockaway, NY: Cummings and Hathaway, Publishers.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science. 7*, 155-170.

Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology, 12*(3), 306-355.

Extended Question/Answer Dialogs in HyperMedia Systems

G. HOLWEG
Institute for Information Processing and Computer Supported New Media (IICM),
Graz University of Technology, Schieszstattgasse 4a, A-8010 Graz, Austria

HyperMedia and MultiMedia enjoy growing popularity as a means for all kinds of presentation, in reference books, games and finally in education and tutoring as well. However, most systems lack functions known from CAI-programs, i.e. the possibility to test acquired knowledge with the help of questions and to judge the answers - either in order to give the learning person some feedback on his/her learning progress or as a base for statistical analysis. Besides conventional methods (such as text input with a following comparison between a predefined answer and a student answer; vocabulary training; multiple choice questions) the use of new media (graphics, animation, software movies, music) makes it possible to create not only new types of questions, but also to react to the answers in a new way. Thus the answer to a question can either be a click on the button, the movement of graphics or even the playing of an instrument with a MIDI-interface. In such a case, judging single input like a mouse click will hardly suffice; hence complex flow structures have to be defined which are then compared with the input of the learning person. To give an example consider a machine which has been taken apart. To reassemble the machine each component must be moved to its correct position on the screen. The order of assembly is not arbitrary (a nut for example cannot be fixed before a screw is set), nevertheless there is still a large amount of possibilities of finding a correct solution (e.g. some other screw can be fixed first). Owing to this, the definition of the flow structure consists of a number of rules such as the following: screw1 -> screw2 -> screw3; nut after screw; screw1 or screw2.

At the present time, experiments are carried out with a prototype of Question/Answer (Q/A) dialogs at our institute. This system will subsequently be embedded into two HyperMedia-systems currently being developed at the institute - the PC-based Hyper-M and the Unix-network oriented Hyper-G. The Q/A-system consists of a number of *interactors* that have the possibility to react to external actions (text input and judgment of student answers, mouse clicks, mouse dragging, time-out by the system clock, run off counters etc.). The whole control of the Q/A-system is done by the *control unit* which prescribes the solutions to a problem as we have seen in the example above. The definition of a Q/A-dialog is done by a script language which will be exchanged by an editor with graphical user interface in the next implementation phase.

In the design of the Q/A-system especially the requirements of Hyper-M and Hyper-G were taken into account. Some of these requirements are listed in the following: The Q/A-system must be topic independent (and can therefore not be in the field of ITS); access to extendible libraries of learning strategies (form filling, multiple choice, drill&practice etc.); different kinds of feedback are solely limited by the HyperMedia systems (software movies, sound, animation etc.); switching between different languages should be possible (English, German, French etc.); easily usable (undo, help how to use the system and how to solve the problem, recording the student's actions, time-outs); use of hypertext features (if you do not understand a word in a question or feedback click on it and the meaning of the word will be searched from an encyclopedia in the background); access to data bases (e.g. vocabulary data bases).

A precondition for the application of some of the features listed above is an access by the Q/A system to the powerful features of Hyper-M and Hyper-G. This will be the next step in the implementation.

Expert's Views About Key Issues for Courseware Development in Advanced Computing Environments

MARY E. HOPPER
Department of Curriculum and Instruction
Purdue University, West Lafayette, IN 47901, USA

New attempts to implement computers in education face many problems faced by older initiatives, as well as some that are a function of newer technology. Overemphasis on advice from experiences with older technology might endanger new efforts, and a cycle of wasteful failure and missed opportunity could result. The interactions of older problems with new technology can be explored by studying environments similar to those that are likely to become more available to education in the future. These environments have the following features:

- Networking systems that support two or more types of workstations or personal computers
- Easy access to international electronic networks (Internet)
- Software for constructing simulations and distributed databases of "linked" text and graphics.

The following organizations have experimented with courseware development in advanced environments, and their efforts can be analyzed with hindsight by their participants:

- Athena, Massachusetts Institute of Technology (Champine, 1991)
- Institute for Research in Information and Scholarship, Brown Uuniversity (Yankelovich, Meyerowitz, & VanDam, 1985)
- Educational Research and Information Systems, Purdue Univeristy (LeBold, Hopper, Feghali, 1991).

The data for this qualitative study was gathered through "ethnographic" interviews with twenty key developers and managers from the above organizations (Spradley, 1979). In addition to transcriptions of interviews, published documents were used as a supplemental form of data. Analysis of the data revealed consistent relationships between three areas of concern inherent in the nature of "educational computing projects". These three areas were pedagogy, technology and organizations. Pedagogical decisions presented technical problems, that resulted in organizational challenges. In attempts to take advantage of valuable pedagogical goals afforded by new technological capabilities, technical challenges became intertwined with traditional organizational structures for distributing and managing human, technological, and financial resources. Over the course of projects, opportunities and limitations afforded by the pedagogical, technical, and organizational contexts became reflected in the character of the courseware that was developed.

References

Champine, G.A. (1991). MIT Project Athena: A Model for Distributed Campus Computing. Bedford, MA: Digital Press.

LeBold, W.K., Hopper, M.E., Feghali, A.A. (1991). A Hypermedia Solution to a Hyper Problem: Personalized Computer Engineering Career System. 1991 ASEE-IEEE Conference Proceedings , p. 482-488.

Spradley, J.P. (1979). The Ethnographic Interview. New York, NY: Holt Reinhart & Winston.

Yankelovich, N, Meyerowitz, N. & VanDam, A. (1985). Reading and Writing the Electronic Book. IEEE Computer, 18(10) 85, 15-30.

Current Designs in HyperCard: What can we learn?

Richard Jacques, Blair Nonnecke, Diane McKerlie†, Jenny Preece

Computing Department, The Open University, Milton Keynes, UK
†Rank Xerox EuroPARC, Cambridge, UK

There is considerable interest in the use of hyper-systems (such as hypertext and hypermedia) for education and learning. We take the position that understanding current design practice can inform us of what this learning potential might be and how it might be harnessed.

We used a usability inspection method, Heuristic Evaluation, to study 16 HyperCard stacks. Design features were identified which would support the users and those that would cause potential problems. To help structure the evaluation, six criteria were identified: **purpose** or author's intent of the hyper-system, domain or **content** information, **structure** of the nodes and links, **navigation** methods, **control** mechanisms, and style of **presentation**. These criteria were chosen for their relevance to both learning and usability issues. Our findings are summarised as follows:

Purpose: Most of the stacks were intended to teach users. We identified a number of more specific purposes: tutorials, demonstrations, experiences and arguments. However, not all conveyed their purpose well. We found that stacks which adequately explain their intentions and learning objectives through the appropriate use of titles and first cards were more successful. This improves the user's chance of being engaged and of positioning themselves with respect to the content.
Content: The contents of the stacks were immensely varied. No matter how interesting the content was to individuals, there was agreement that motivation to continue was strongly linked to structure and presentation.
Structure: The structure of the stacks varied from simple loops of cards to complex hierarchies. Mental models of loop structures are easier to form than mental models of hierarchies. However, loop structures were slower to use, more cumbersome and less adaptable.
Navigation: Stacks arranged in loops based their navigation on simple 'next' or 'previous' arrows, while hierarchies used *table of contents*, *search*, *find* and *map* tools. The most successful were those that not only showed the user how to get to links, but provided them with sufficient guidance to make informed decisions.
Control: Stacks that allow users to do what they want to do (i.e., are flexible but supportive) provide positive learning experiences.
Presentation: Presentation style (i.e., choice and combination of media) is often the aspect that is first noticed. The stacks that did make a good initial impression engaged and motivated us to continue. Surprisingly, many of the stacks exhibited poor uses of typography and graphical design.

In conclusion, good hyper-systems have significant learning potential, but their design and development requires much consideration. It is imperative that authors ensure that users will understand the aim or purpose of the system; the way the content is structured; and how to access, control, and navigate through the information. Users will learn best with hyper-systems that meet their expectations, engage, and motivate them.

ERIC

Multimedia Captioning: Synchronized Text and Audio Presentations

Dr. Cynthia M. King
Gallaudet University, Educational Foundations and Research
800 Florida Avenue, NE, Washington, DC 20002-3695

The future is bright for multimedia (new media or interactive media, depending on your choice of terminology). Almost all projections (e.g., Reveaux, 1993) indicate rapidly growing markets for products that combine text, audio, graphics, animations, and still and motion video with interactivity and user-control.

Imagine, however, that you are deaf. Turn off the sound in your favorite multimedia product and imagine what it would be like to live in a world where your primary link to the world was visual, rather than auditory. Would you be able to "hear" the instructions in Broderbund's *Just Grandma and Me*? Would you have access to the speeches and narration in *Clinton: Portrait of Victory*? How much information would you get from the vast number of videodiscs that are not captioned?

Captioning, the process of providing a synchronized written script (captions) to accompany auditory information, is the focus of this paper. During the presentation at the ED-MEDIA Conference, various methods of multimedia captioning will be presented, along with demonstrations of problems (and some solutions) regarding uses of standard television captioning in multimedia products. This paper defines multimedia and television captioning and provides answers to three important questions:

1. How much of a problem is media accessibility for deaf and hard-of-hearing individuals?
2. What are the benefits for me, as a multimedia developer, if I include captioning as part of my multimedia development efforts?
3. What is needed to do multimedia captioning?

Definitions

(Digital) multimedia captioning involves the use of digital objects (e.g., text fields, graphics, animations, or motion video) as mechanisms for providing visual representations of non-visual information. Captioning differs from titling in that captioning involves the synchronized presentation of audio and visual information, whereas titling does not involve extensive synchronization and is most frequently used for titles, dates, and other information. Multimedia captioning is applicable in computer-based systems and the digital television systems of the future.

(Analog) television captioning involves (a) the encoding of data into Line 21, field 1 of an NTSC's television signal's vertical blanking interval, and (b) decoding of that data (resulting in display of captions on a television screen) via a decoder in the viewer's home (Armon, Glisson, & Goldberg, 1992). This system of captioning is commonly known as

closed-captioning in that the consumer must have a decoder to view the captions (i.e., captions are "closed" to those without a decoder). As of July 1, 1993, all televisions (13" or larger) built for sale in the United States will be able to decode closed-captioning without an external decoder.

How much of a problem is media accessibility for deaf and hard-of-hearing individuals?

Although great strides have been made recently in providing accessibility to communication technologies such as telephones, televisions, and telecommunications (e.g., Harkins, 1992; Jordan, in press), accessibility to computers, and especially multimedia products, remains a major problem for most deaf and many hard-of-hearing people.

For example, less than two percent of programs (28 out of 2,200) listed in the *VideodisCompendium for Education and Training* (Emerging Technology Consultants, 1993) are identified as being closed-captioned. The situation is somewhat better for entertainment video available on videodisc where approximately 17% are identified as closed-captioned (i.e., 1,200 out of more than 7,000 videodiscs listed in the *Laser Video File*, Fall, 1992/Winter, 1993). Still, a majority of materials on videodisc remain inaccessible for independent use by people who rely on vision, rather than audition, for communication.

Even those videodisc materials that are closed-captioned can become inaccessible when the video is used in an interactive environment. Non-linear presentations of video can cause the captions embedded in analog video to become scrambled, disappear temporarily, or disappear permanently until the decoder is reset (King, 1992a, 1992b). Further, closed-captions sometimes overlap shot changes or are not timed optimally (due to data transmission constraints in the system, Armon et al., 1992). These constraints cause additional difficulty when analog video is used in an interactive manner (King, 1992b). Some of the problems described above can be remedied by "fooling" the decoder via computer-control of the videodisc or creative programming within the multimedia environment (King, 1992b). Other problems, especially those associated with CLV-type videodiscs, are more resistant to remedy (King, 1993).

Even if all of the problems were solved and all videodiscs were closed-captioned, however, it would not mean that accessibility to multimedia would improve for deaf and hard-of-hearing people. The future is digital video, not analog (e.g., Grunin, 1993; Rosenthal, 1993). As Reveaux (1993) notes, videodiscs and other analog media may have limited futures in terms of their role in computer-based multimedia products.

Much of the evidence concerning accessibility problems with digital media is anecdotal (King, 1993). The breadth and severity of the problem, however, is easily imaginable by persons familiar with the rapidly expanding market for digital media. The ease and relatively low cost of creating digital audio, by itself, creates a potential nightmare for the computer user who is dependent on visual means of communication. Heeter and Gomes (1992) offer a projection of what the future might hold: "it is time to fully integrate sound into computing environments; that someday we will look back on the anachronistically silent word processors and spreadsheets of today in much the same way that we appreciate a silent movie for its historical value" (p. 155). Is it possible that interactive multimedia, which has such potential for visual-oriented people, will be a barrier to full and equal communicative access for deaf and hard-of-hearing people?

What are the benefits for me, as a multimedia developer, if I include captioning as part of my multimedia development efforts?

As a multimedia developer, you might be asking yourself, "What does this have to do with me? I don't develop for deaf or hard-of-hearing audiences." There are several reasons that you should consider integrating uses of captioning into your multimedia products.

First, the American with Disabilities Act of 1990 requires reasonable accommodations for disabled populations (e.g., Castorina, 1993; Harkins, in press; Lazzaro, in press). As Harkins (in press) suggests, developers who consider accessibility issues in the design of their products will have a competitive edge in sales. Further, "if developers of technologies are aware of and take into consideration in their designs the needs of individuals with disabilities, the designs will be more cost-effective in the long run" (Schiller, 1993, p. 6).

Second, evidence is accumulating that captioning can be helpful to a variety of audiences. DuBow (1991) noted that 40% of the caption decoders sold in 1989 were sold to people for whom English is a second language. A multitude of research studies have also shown positive effects on literacy for a variety of disabled and non-disabled populations, including adult non-readers, non-disabled children, learning-disabled children, and individuals learning English as a second language (see, for example, the NCI *International Bibliography of Captioning and Subtitling Literature*, 1991).

Third, as with other assistive devices designed for special populations (e.g., access curbs for wheelchairs), the general population has already begun finding other uses for captioning. Museums, for example, sometimes caption interactive or video displays that are used in public places (reducing noise and eliminating the need for headphones).

What is needed to do multimedia captioning?

Three conditions are necessary before systematic inclusion of captioning in multimedia products can occur: (a) provision of a captioning capacity within digital/multimedia environments, (b) tools for creating multimedia captions, and (c) extensive use of captioning capacities and tools. Progress is being made in meeting each of these conditions. [During the ED-MEDIA Conference, the capacities and tools will be demonstrated.]

<u>Provision of Captioning Capacity</u>. In the object-oriented environments in which multimedia captioning takes place, captions are simply standard objects that can be manipulated like other objects; thus captioning can be easily accommodated. IBM's operating system, *OS/2*, already has a built-in captioning capacity, and many multimedia programming environments (e.g., Apple's *Hypercard* and Asymetrix's *Toolbook*) can be programmed for synchronized presentation of text and audio. Finally, captioning capacities for digital video formats (e.g., *Quicktime*, *Video for Windows*) are also becoming available or are in development.

IBM's captioning capacity in *OS/2* can be used as a model for others. *OS/2* provides a toggle in MMPM (Multimedia Presentation Manager) that determines whether or not captions are displayed. This concept, based on the closed-captioning system from television, is an important one in that users can choose whether or not they wish to see the captions. Too, incorporation of the capacity into the operating system ensures consistency and reduces the need to build captioning capacity into each program used for multimedia development.

<u>Tools for Creating Multimedia Captions</u>. Tools for titling and captioning are also becoming available. Various desktop video editing systems (e.g., Matrox's *Illuminator* on the PC; Avid's *Media Composer Pro*) have video titling tools built into them. Such systems, which typically require manual insertion of each title, can be used for captioning. Most developers, however, will find them too labor-intensive.

Captioning tools, analogous to WGBH-TV's *CC-Writer* and other similar products for television and video captioning, are needed. Doug Short's *SyncText* (Institute for Academic Technology, 1992) is perhaps the most feature-rich set of tools for multimedia captioning currently available. These tools were developed using Asymetrix's *Toolbook* on the PC as a means for providing English translations for foreign operas on videodiscs and for annotating audio CD's with explanations or musical notations. Like *OS/2*, *SyncText* might be used as a model for development of similar tools for other environments.

<u>Extensive Use of Captioning Capacities and Tools</u>. Captioning capacities and tools, however, will be worthless unless developers use them to incorporate captioning in their multimedia products. The involvement and support of multimedia developers--<u>YOUR involvement and support</u>--is the key to ensuring that deaf and hard-of-hearing people are not denied full and equal access to what many of us believe is the most exciting development of the century: interactive, multimedia learning environm[illegible].

Summary

The purpose of this paper was to provide those in leadership positions (the audience at ED-MEDIA and members of the Association for the Advancement of Computing in Education) with greater understanding of accessibility issues, other reasons for using captions, and some beginning knowledge of how captioning can be done in multimedia environments. Hopefully, it has accomplished those goals and given readers incentive to explore multimedia captioning.

King and her colleagues at Gallaudet University, among others, will be continuing to explore accessibility issues and developments in multimedia. Individuals wishing more information can contact her at the address at the top of this paper or via electronic mail (cmking@gallua.BITNET or cmking@gallux.gallaudet.edu).

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. <u>SMPTE Journal</u>, <u>101</u>(7), 495-498.

ERIC

Castorina, C. (Ed.). (1993). *Opportunities: Equal access to electronic library services for patrons with disabilities*. Washington, DC: EDUCOM, The Interuniversity Communications Council.

DuBow, S. (1991). Television Decoder Circuitry Act: TV for all. *Temple Law Review*, *64*, 605-614.

Emerging Technology Consultants. (1993). *Videodiscompendium for education and training: Laserdiscs, CDs, and multimedia software*. St. Paul, MN: Author.

Grunin, L. (1993, May 11). Desktop video: A window on the future. *PC Magazine*, 12(9), 289-292, 294, 302-304, 306-308, 310, 312-313.

Harkins, J. E. (1992). *Technology and deaf children*. Testimony presented to the United States Congress.

Harkins, J. E. (in press). Ergonomic considerations for communication technologies for deaf and hard-of-hearing people. *Proceedings of the Fifth International Conference on Human-Computer Interaction (Orlando, FL; August, 1993)*. The Netherlands: Elsevier Science Publishers.

Heeter, C., & Gomes, P. (1992). It's time for hypermedia to move to "talking pictures." *Journal of Educational Multimedia and Hypermedia*, *1*, 255-261.

Institute for Academic Technology. (1992, Fall). Faculty develop programs for listening comprehension and text synchronization. *IAT Briefings: The Institute of Academic Technology*, pp. 14-15.

King, C. M. (1992a). *Project ALIVE!: Acquiring Literacy through Interactive Video Education*. Grant funded by the US Department of Education, Office of Special Education and Rehabilitative Services. [Grant #180H20034]

King, C. M. (1992b). Captioned and uncaptioned videodiscs: Problems and solutions. Paper presented at the 1992 National Symposium: Educational Applications of Technology with Deaf Students, Rochester, NY. [Abstract to be printed in the *American Annuals of the Deaf*, 1993].

King, C. M. (1993). Project ACCESS: Tools for Equal Access to Multimedia. Grant submitted to the US Department of Education, Office of Special Education and Rehabilitative Services.

Jordan, I. K. (in press). Colloquium: Access to technology for deaf and hard-of-hearing people. TechNos.

Laser Video File. (1992, Fall/1993, Winter). *Laser video file: the complete list of laser video discs*. Westwood, NJ: Author.

Lazzaro, J. (in press). *Adaptive technologies for learning and work environments*. Washington, DC: American Library Association.

National Captioning Institute. (1991). *An international bibliography of captioning and subtitling literature*. Washington, DC: Author.

Reveaux, T. (1993). Title wave: CD-ROMs find a market. *New media: Multimedia technologies for desktop computer users*, *3*(5), 36-40, 42, 44.

Rosenthal, S. (1993). It's showtime: *Quicktime* vs. *Video for Windows*. *New media: Multimedia technologies for desktop computer users*, *3*(3), 36-39.

Schiller, E. (1993, February 18). *Agenda for the Technology, Educational Media, and Materials Program for Individuals with Disabilities*. Washington, DC: US Office of Special Education Programs.

ERIC

RHINO: Designing Multimedia Curriculum Blocks

Miriam J. Salman Masullo
Thomas J. Watson Research Center
Yorktown Heights, New York 10598

This paper focuses on the use of multimedia for representing, examining and modifying curriculum by educational professionals in a teacher support system. In order to illustrate the concept, a live presentation of several multimedia curriculum blocks isolated from the RHINO Model (Masullo, 1993) will be demonstrated in a workstation.

A curriculum is organized into multimedia curriculum blocks. The elements of these blocks vary depending on the intellectual content of the block. However, the common elements that relate to all the blocks are: educational concepts and principles, recommendations on appropriate resources, suggested activities to re-enforce the content, and related forms of assessment. These are categories of information in each curriculum block, and they correspond to well known fundamentals teachers consider when planning lessons, such as skills, concepts and attitudes.

Concepts and principles: refer to the major understandings that are fundamental such as properties and states of matter for a physical science block.
Resources: wo ıld provide references to related materials, from text to video, to software.
Activities: p ıray actual experiences in the classroom, or in other settings in video form.
Assessment: are suggestions or recommendations in the form of voice annotation by teachers who have had experiences either with the resources or the activities, or who specialize in the content area.

The multimedia capabilities of this form of curriculum modeling can be useful for teacher support as well as for teacher training. Teachers can observe actual groups taking part in an activity or get information about authentic assessment that is relevant to that particular activity. Teachers can update their skills by exploring the information while watching a video demonstrating how to conduct an experiment, for example, or how to use a new computer-based tool. At the present time, teachers must attend workshops or read about tools or ideas. These forms of support are not always available or are not a possibility when needed, while with this relatively modest approach a teacher could explore a descriptive block at any time and for any number of times.

More research is needed to understand and explain categories of information within multimedia curriculum blocks, and to define categories of multimedia curriculum blocks. Research, already in progress, is addressing the issue of linking these blocks in a database-like fashion. The collection of blocks represented in this manner will be used to formulate a large hypermedia curriculum database. Additional issues under investigation include the management of a large set of multimedia curriculum blocks and a mechanism for filtering blocks through standards and other forms of accountability criteria. Schemes that could be used to facilitate selection of blocks by educators are under study as well. In conclusion this approach to large scale curriculum modeling using multimedia, generates many interesting research questions.

[Masullo 1993] Masullo,M. *RHINO: A Curriculum Support System, Creating Learning Experiences* NECC-93, Orlando, FL., 1993.

628

ERIC

ArchiMedia: Interactive Architecture

MARIETTA S. MILLET, PETER COHAN, GRANT HILDEBRAND
Department of Architecture, JO-20
University of Washington
Seattle, WA 98195, USA

ArchiMedia is an interactive reference tool that is designed to be used by students of architecture in the design studio. ArchiMedia is being developed as a multimedia tool to assist students and teachers in that exchange of information that feeds the process of integration so essential to architectural design. ArchiMedia presents case studies of significant buildings in an dual way, providing both "soft" images to evoke the functioning of the right brain, and "hard" images to stimulate the left brain.

Several ideas inform this approach. Architecture, as the combination of many aspects, both technical and artistic, cannot be understood in a simple one-dimensional way. Context (historical, cultural, physical, economic); structure, materials, and construction; patterns of experiential awareness ("procession", "mystery", "prospect and refuge"); formal patterns; thermal, auditory, and luminous experiences; the architect's experience and approach - all feed into the complicated totality that is a building.

But students - and even teachers and practitioners to a large extent - are exposed to only partial information about any one building. For instance, one building may be lauded in history class, and another extolled in structures, and yet another praised by design studio teachers, all for different reasons. In reality, all buildings have a complicated genesis that, when analyzed and understood, puts the many aspects of "architecture" in perspective. ArchiMedia follows an abbreviated Case Study format to make available a broad selection of information about selected buildings.

The interactive format of ArchiMedia allows the user to plot their own unique course through the material, making selections through association. With a combination of the software capability and strict guidelines for formatting, a wide range of connections can be made, for example: buildings in a certain a time period, or location; buildings by one architect; buildings of one structural type, or material; buildings with the same experiential pattern, such as "mystery"; buildings with the same organizational diagram or parti; or a certain genre of buildings, such as museums. The innovation lies not only in gathering the material together in one place for simultaneous access, but also in carefully structuring the connections to allow real freedom of use in an organized way.

Developed in Supercard, ArchiMedia requires only a Mac-II platform for operation and displays all information and images in high-resolution color on the screen. This technology opens up unique and stimulating ways of manipulating graphic information, enhancing the comprehension of ideas essential to architecture. Side-by-side and superimposed comparisons of building plans and sections, three-dimensional color-coded renderings, and animated sequences create presentations unavailable in printed form. The use of numerous high-quality color images in ArchiMedia is critical to understanding a building , and yet color images are limited in printed matter due to cost.

Opportunities for networking are presented in the project. A venture of this magnitude requires input from many sources as to content and images. The schools of architecture around the country and world possess these resources. By sharing these resources, the project gains depth and breadth and quality and the contributors gain access to ArchiMedia.

Using Multimedia to Teach Thinking and Problem Solving Skills

NANETTE MITCHELL and SUZANNE SUPRISE
Rollins College, Winter Park, Fl 32789, USA

Technology of the 1990's has created unparalleled teaching possibilities, making the development of multimedia applications in all disciplines a relatively easy task. But we see this powerful media as a way to teach more than just content; rather, we envision a system of modules that teach thinking, or reasoning, skills--those skills necessary to solve problems in the *real world*. The different, but integrated, processes of problem solving involve several levels of complexity and eventually culminate in application, or the connecting of acquired knowledge to contextual demands. As such, problem solving is an interactive process in which the learner sifts through information, picks out critical features, then compares and constructs relationships among the information to make decisions.

To build such component skills, educators must adhere to teaching principles such as (a) directing attention to the critical features of a problem, (b) presenting material repeatedly in varied contexts, (c) guiding the learner through successive steps of complexity, (d) providing opportunities for practice with immediate feedback, and (e) making connections between new information and previously acquired knowledge. Multimedia can be an effective vehicle for incorporating these teaching principles through the purposeful integration of video, graphics, animation, and sound, and through the strategic application of behavioral principles. By developing interactive modules that augment classroom presentations and provide the learner an opportunity for independent exploration and practice, learners ultimately acquire such component skills as sifting, discriminating, comparing, and evaluating problem information.

In interactive computer-based modules, problems can easily be presented in contexts that are familiar to the learner by incorporating varied and relevant video clips of *real-world* events, and by subsequently revealing the salient features of the problem through spoken or written descriptions. Learners are first presented simple problems containing visual illustrations, supporting materials, and other cues to guide their thinking, and then they are given an opportunity to manipulate variables of a problem and immediately see the effects via animated graphics which simulate the experience. Learners progress through stages of increasing complexity simultaneously with a decreasing degree of explanatory support. Activities also require increasing levels of interaction to promote the generalization and application of skills.

Multimedia offers more than a novel means of conveying content; it is a powerful educational tool that can deliver an individualized presentation of information that is difficult to achieve in the classroom, or even in a laboratory setting. It allows the learner to construct and test varied solutions to a problem by interacting with data. Through this interactivity, learners actively acquire the thinking skills necessary for academic success.

ERIC

Pro'olem Solving in a Multi-Media Environment

HENRY F. OLDS, JR.
Senior Scientist
Bolt Beranek & Newman, Inc., Cambridge, MA 02138, USA

Tra ' tional teaching of problem solving has generally abstracted problems from the real world and simplified them into supposedly meaningful and understandable chunks. From the student's point of view, the result has often been problems that appear capricious, trivial, and often silly. From society's point of view, the result is a very large number of students who are unable to solve problems in their work or in their lives. With multimedia computer technology, it is now possible to put in our classrooms rich problem solving environments and the powerful mind-extending tools that enable students to define and solve the many complex problems that emerge. As a result, it may be possible to change the very nature of the educational process.

Copley Place is a unique, multi-media, computer program, created as a design example to engage students in extended inquiry into a complex problem. It is intended to teach interdisciplinary problem solving skills to students in grades from 6-10. But, more important, it is intended to teach students how to approach and gain understanding of complex problems and how to use various languages (mathematical, symbolic, verbal, graphic) for communicating that understanding. Much as process writing has become a widely accepted approach for helping students sharpen their verbal thinking skills, Copley Place encourages "process solving" as an approach to sharpening problem defining and solving skills and solution presenting abilities.

The curricular approach of Copley Place engages students in an interdisciplinary investigation of a large, complex real problem -- the construction of a $500,000,000 multi-use, urban, redevelopment project. The program "drops" students into the middle of the most significant problem faced during the project: how to help Tom Perkins, the real superintendent of construction, construct on unstable filled land a foundation that can support a 31 story modern hotel. This is the focal problem that sits at the center of a radiating set of related interdisciplinary inquiry paths, which are available to students should they wish to pursue them: a path that leads back in history to the time when the land on which the project is being built was tidal marsh and was being filled; a path that relates the protest made by the community to the failure of the project to deal adequately with the local need to replace housing torn down by the project; a path that explores problems associated with ordering, scheduling, and installing interior treatments and furnishings for the completed hotel; a path that explores the economics of operating a small theater complex in the heart of the shopping mall; and a path that explores the design and installation of a large waterfall sculpture in the middle of the shopping complex.

In addition to the video imagery that connects students to the context, the program includes tools for representing particular aspects of the problem in visual formats that lend themselves to manipulation and analysis. Using such tools, students can arrive at solutions that are not only correct but that are understood. For example, on the computer, the student can create an abstract representation of the volume of the hotel's foundation and then figure out how to express that volume in terms of the number of cubic yards of concrete needed to fill it and then the number of truckloads and amount of time it will take to get the concrete to the site.

Using Copley Place for education poses multiple challenges for typical classrooms in typical schools. Short school periods do not lend themselves well to the kinds of projects that Copy Place engenders. Working through the projects suggested by Copley Place can take many days, possibly weeks, of precious class time. Because it is not always clear what right answers are, teachers and students may have difficulty agreeing about what constitutes quality products. However, out of such challenges can also emerge a clearer understanding of why and how it is important to break the educational molds that constrain today's school from providing the conditions that lead to world class education.

ERIC

Meta-analysis: The Effectiveness of CAI in Elementary Education

JOHN OUYANG, GAIL GERLACH
GEORGE BIEGER, VINCENT MIKKELSEN
Department of Professional Studies in Education
Indiana University of PA, Indiana, PA 15705, USA

Following Glass' introduction of meta-analysis into the field of social science research in 1976, a number of researchers have turned their attention to reviewing the effectiveness of CAI using meta-analysis. However, only few previous meta-analytic reviews have focused on the effectiveness of CAI in elementary education, and few of them have reviewed the studies conducted after 1986.

For the purpose of providing up-to-date information of the effectiveness of CAI in elementary education (K-6), this study reviewed 79 individual studies conducted after 1986 using Glass's meta-analysis. Among the 79 individual studies, 35 were published journal articles, 32 were dissertations and 12 were ERIC documents and unpublished papers. In total, 267 effect sizes were calculated and analyzed through SPSS.

The results of the present study showed that children in the elementary schools raised their achievement scores by 0.495 standard deviation units when CAI was used. Among the academic subjects taught with CAI in the schools, there was a significant difference between the instruction of spelling (ES=.85) and reading (ES=.16). Among the children using CAI in elementary schools, the higher effect sizes were found for the children in the intermediate grades (4-6) (ES=.57) than those in the primary grades (K-3) (ES=.42). Only two of 267 effect sizes with an average of .30 was related to the multimedia CAI. However, no superiority of any application of CAI were found. As the instructional time lasted longer, the effect sizes tended to become lower. There was a significant difference between the instructional time within four weeks (ES=.67) and the time of over fourteen weeks (ES=.32). The results also showed that the higher effect sizes were related to the most recent years' studies (ES >.50).

Based on the statistical analysis, it can be concluded that 1) CAI seems to have a positive effect on improving children' academic achievement: 2) Some academic subjects such as spelling may be taught more effectively than others, because the nature of learning only needs a paired association (Niemiec, Samson, Weinstein, & Walberg, 1987); 3) Children in the intermediate grades appears to benefit more from CAI than those in the primary grades, which confirms that as children grow up, the relationship with computers is developed from meta-physical stage to mastery and identical stages (Turkle, 1985); 4) More research on multimedia CAI in elementary schools is needed in the future to see its virtue advantages: 5) Children's novelty using CAI should be motivated and further explored; and 6) Newly developed CAI hardware and software may have an impact on the effectiveness of CAI for children in the elementary schools.

References

Niemiec, R., Samson, G., Weinstein, T., & Walberg, H. (1987). The effects of computer based instruction in elementary school: A quantitative synthesis. *Journal of Research on Computing in Education, 20*(2), 85-103.

Turkle, S. (1985). *The second self: Computers and the human spirit.* New York: Simon & Schuster.

Interactive classroom presentations and whole group fee ·back

KAREN SWAN
Department of Educational Theory and Practice
SUNY Albany, Albany, NY 12222, USA

Educators have long understood that the most effective learning situations provide opportunities for overt student response. The positive effects of individualized feedback have been demonstrated in a wide variety of areas, ranging from perceptual skills (Snoddy, 1926) through memory (Underwood, 1957) to problem solving (Newell & Rosenbloom, 1981). It has been extremely difficult and expensive, however, to integrate feedback opportunities for all students in classroom settings. Until now. The same computing technologies used to develop interactive learning environments for individual student use are now being enlisted to create classroom presentations that incorporate the active participation of all students. While whole group feedback retains some of the attributes of individualized feedback and might logically be assumed to entail many of its benefits, the whole group presentation situation requires that instructors respond directly to student feedback, and that students extract from such responses the information applicable to themselves. This study investigated learning in just such a situation.

Methodology. Subjects were 48 graduate students of education randomly assigned to one of two groups. Subjects in both groups were given a computer-based, whole group, multimedia presentation on formal video conventions that consisted of their being shown sets of definitions of video conventions together with video clips illustrating each convention. Following each of the sets, subjects in the experimental (feedback) condition were shown a second series of video clips and asked to identify them using individual keypads connected to the computer through a student response system which displayed student responses as a histogram. When such display indicated a sufficiently large proportion of the class (> 25%) incorrectly identified a convention, the instructor explained the correct answer. Subjects in the control (no feedback) condition were given identical instruction and shown the same video clips as experimental subjects The only difference between the two groups involved the presence or absence of whole group feedback. Immediately following instruction, all subjects were tested on their knowledge of video conventions. The test consisted of two parts -- a definitions section (15 questions) and an identification section (10 questions). Mean overall test scores and mean scores for each section were compared between groups using single-tailed independent t-tests.

Results. Subjects in the experimental group scored 2 points (out of 15) higher on the definitions section, and 1.5 points (out of 10) higher on the identification section of the test than control subjects. Mean overall scores for the experimental group were thus more than 3.5 points (out of 25) higher than those of the control subjects. The results of the independent t-tests indicate significant differences between the mean scores of experimental and control subjects on both sections of the test (definitions; $t=2.75$, $p<.01$; identification; $t=3.26$, $p<.01$), as well as on the test as a whole ($t=2.56$, $p<.025$). Such findings suggest that the experimental group learned better as a result of the interactive sessions they received. The use of a single computer linked to student response units is a far less expensive means for providing feedback opportunities than the use of individual workstations. If whole group feedback is found to be even half as effective as individualized feedback, student response systems might be used to enhance traditional instruction in an affordable way. The technology clearly deserves further investigation.

A FULLY INTEGRATED ELECTRONIC DISTANCE TEACHING MODEL FOR THE UNIVERSITY OF SOUTH AFRICA

DR P. HENRY R. VAN ZYL
Department of Educational Technology
University of South Africa, Pretoria, Republic of South Africa

Computer technology in distance education does not refer simply to the use of computers on a wider scale, but to fully integrated systems, based on sound didactic principles, designed to promote effective teaching, enabling students to learn effectively according to preferred learning styles. A paradigm shift is necesary: from a teacher centred model to a student centred model aimed at promoting independent, critical, problem solving learning skills.

In distance education, students should be allowed to register and to take examinations according to individual needs. This implies a) an open system, exploiting electronic education and learning in co-operative learning opportunities, and b) adopting a totally integrated multi-media approach, where supplementary media contribute materially to students developing creative problem solving approaches.

Permanent exam centres, where test and exam papers are electronically generated on site are also necessary. These centres can provide a full range of electronic educational opportunities, taking these media and facilities to the people. Once again, the student is the focus: better opportunities must be created to hone their skills as independent knowledge workers rather than submit to the temptation to churn out vast numbers of students who have merely learnt how to answer exam questions more or less successfully.

Students using smart cards must have selective read-only access to all required information via numeric keypads such as those at ATMs - this could become a reality overnight if an agreement is reached with the banks providing countrywide ATM networks. Linkups with the computerised library catalogue will enable students and lecturers to request any loan items from the library, access CD ROM material or browse through electronic book indexes. Online linkups with the University's editorial department will enable study guides and tutorial letters to be accessed from electronic bulletin boards and to be printed at home.

Video conferences now take place between the Pretoria campus and Cape Town, and will be extended to other centres. Synchronous and asynchronous computer conferencing can be made available to students worldwide, promoting and enhancing critical thinking skills and independent learning as students and faculty take part in continuous academic debate without regard to time or place. Thus a truly open university becomes a distinct possibility where true distance learning is unhampered by any but the most essential restrictions. Ways and means of preventing the lowering of academic standards, and preventing students from being prejudiced in any way, must also be found.

The University of South Africa is already close to achieving many of these ideals. Official policy on information technology, incorporating many of the above proposals, is now being drafted. Much of the technology required is already available, or is being developed. Some teaching departments have already started to move along the lines indicated here. What remains is to change the attitudes of staff so that they will not resist change, but welcome it and exploit the greater possibilities it offers for students to develop into independent knowledge workers rather than to remain passive recipients of a one way teaching traffic, ultimately benefiting those in whose service the University stands: the student population.

ERIC

A Tutoring-text Model Based on a Multi-layers Network

Toyohide WATANABE, Atsushi TANAKA and Noboru SUGIE
Department of Information Engineering, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-01, JAPAN

The tutoring-text structure must not only accommodate domain knowledge, but also indicates the characteristic relationships among learning units with respect to the support of effectual tutoring process. The requirements are satisfied with our hierarchical network structure. We call this structure as the tutoring-text model. The hierarchical structure has an effect with respect to making the construction/management of domain knowledge easy, while the network structure operates tutoring relationships among learning units.

The top layer in this model is a directory of domain knowledge, and in the bottom layer the nodes are distinct learning items. The learning unit in the higher layer is composed of several related learning units in the lower layer, like subtrees. The learning units in the bottom layer are atomic elements of concretely specified contents, and are called learning items in case of distinguishing these from other learning units explicitly. The learning units are frames to control learning process and manage learning components. Our learning items are classified into explanation items, example items, exercise items, figure items and table items, according to the constructive contents. Usually, the explanation items take main roles of the learning items against the other items. Although these items are collected as individual learning units, they are independently connected through various kinds of links.

The links characterize the connective relationships among learning items/units. The links among learning units represent that all learning items have the same relationship for another learning unit. These links for learning items/units are applied recursively from layer to layer. First, we explain links among learning items. (1) needed-knowledge link(N-link): This is adaptable to all learning items, and indicates the instruction order set among different learning items partially; (2) target-knowledge link(T-link): This is available from an example and/or exercise item to an explanation item, and represents which explanation item the example/exercise item is derived from; (3) similar-example link(S-link): This is set from an exercise item to an example item, and indicates that the same resolution method is also applied to this exercise item; (4) derived-exercise link(D-link): This is available between example and exercise items, and indicates that the resolution method in the example item is adaptable to the linked exercise item; (5) example/exercise-specific link(E-link): This, attached to explanation items, points out an exercise(or example) item; and (6) reference link(R-link): This is adaptable to every learning item, and refers to a figure(or table) item.

Next, we address links among learning units. The links among learning units are fundamentally propagated from the relationships among explanation items. This is because other links such as T-link, E-link, S-link, D-link and R-link are always dependent on explanation item, as one study process. (7) needed-knowledge link(N-link): The meaning is the same as that for learning item; and (8) similar-knowledge link(K-link): This is set among mutually related learning units in each layer, and indicates that they are similar or conflicted in the contents mutually.

Demonstrations/Posters

Traditional Native Storytelling Using Computer -Based Multimedia

DALE AUGER, MICHELE JACOBSEN, BOHDAN J.BILAN
University of Calgary at Calgary, Alberta, Canada

Nine artistically talented youth, from the Wabasca-Desmarais community (600 miles north of Calgary) in Northern Alberta, published interactive multimedia versions of their native stories at a traditional native storytelling workshop in August 1992. A traditional camp was pitched next to the local lake, and a Mac lab was set up in the local school. The focus of the workshop was to bring together native youth, culture and language, and the multimedia capabilities of computer technology. The participating youths ranged in age from 10 - 17 years old, were artistically gifted, and while all of them understood Bushland Cree, only half of them spoke it fluently. These children were schooled primarily in English, and some spoke Cree at home with their parents and grandparents. The children brought with them their stories, and an eagerness to learn more about themselves, their culture and language, and computer technology.

A video and photographic record was kept of the traditional story telling, artistic and computer creation process.

The first two days of the workshop were spent primarily at the campsite, telling stories with elders from the community, drawing pictures with local artists, and writing. The second two days of the workshop were spent in the local school learning, how to use the Macintosh desktop, HyperCard, MacPaint, QuickTime and revising the written stories on the computer, editing their digitized drawings, and publishing interactive multimedia versions of their stories. Our primary goal was to bring together a group of native youth to tell stories, write, and draw pictures about their own culture. We wanted to create an open environment so the children could freely share and write down their Native stories, and learn more about their culture. We encouraged open pride in their native heritage, and tried to reinforce these feelings by making the primary focus of the workshop on traditional native storytelling. We also wanted to include as many traditional elements of native culture into the workshop as possible, because many of these children came from modern families and had not experienced traditionally native ways of life before. Nine stories were published.

ERIC

GREFI, a talking software

René L. Beauchamp
Adult Education
Commission scolaire des Mille-Iles
Laval (Quebec) Canada, H7L 2P1

The "GREFI"software was produced according to the plans of a Research group called "Groupe de Recherche en Evaluation Formative Informatisée (GREFI)" from the Université du Québec à Montréal (UQUAM). This software allows any student to evaluate himself and to receive a prescription at the end of the process if the results did not meet the expectation.

For the teacher, this software is a test editor that allows him or her to see the student's progress getting ready for the final exam.

The Research Group came to the conclusion that a formative testing software should allow each student to:

- with a password, be able to access to his or her file to see the objectives mastered and the one that are left to be mastered,
- be able to choose any objective he or she wishes to be tested on,
- receive a predefined number of questions taken from a bank of questions. The teacher decides the number of questions that must be asked to the student,
- be able to type constructed answers of up to 55 characters,
- receive a reinforcement and a feedback after each answer,
- receive traces and a prescription if the test was failed,
- receive a printed proof if an objective was mastered.

The teacher can effectively ...:

- register the number of groups desired and modify these groups,
- ask questions that can be accompanied by pictures or graphs,
- follow each student's progress (objective mastered and non mastered, the number of time the student had to redo a test, etc.)
- possibly modify for each student the number of objectives to mastered,
- print the results of a group of students.

With this software, it is possible to...

- create a new content, that is a set of objectives, of questions for each objective and a prescription for each objective,
- establish the minimal score,
- modify any content,
- insert graphics, figures and any digitalized pictures,
- print the prescriptions and the questions,
- import questions from an existing bank of questions,
- see and print statistics of students' achievements,
- be able to work on different disk drives.

The "GREFI" software has won the first prize called "APOLOG-91", a contest organized by the Ministry of Education of the Quebec Government. The software is available in French and in English and is sold by Quebit Inc.* A Macintosh version is on it's way.

Quebit Inc., 1584 Fleury est, Montreal, Quebec, Canada, H2C L6S".

ERIC

VideoGraph Development Project

ROBERT J. BEICHNER
Department of Physics
North Carolina State University, Raleigh, NC 27695, USA

The central task of this two year project is to develop improved instructional materials for introductory physics instruction which take advantage of state-of-the-art technology. A curriculum utilizing a visually-oriented approach to kinematics data collection and analysis will be written. The development of the curriculum will be guided by recent findings in cognitive psychology, the work of researchers in science education, and the experience of veteran teachers. My hope is that we will be able to produce materials which can be used to stimulate student interest in science, help them become efficient users of technology, and develop both intuitive and quantitative understanding of the study of motion–a "building block" topic in physics.

The approach I intend to use is based on an extension of earlier work. In a 1987-88 project, I developed a software package called *VideoGraph*. This instructional tool was created for use in introductory physics laboratories at the high school and college level. It allows students to videotape motion events and use the graphing capabilities of a microcomputer to carefully examine and analyze the motion. More specifically, the computer replays the **video** on its screen while simultaneously creating a **graph** of position or speed as a function of time—hence the name *VideoGraph*. This particular use of the computer as a data collection and analysis tool derives from several studies of the successful use of microcomputer-based laboratories. My contention is that by seeing both the concrete motion event and the abstract graphical representation of that motion, students will be better able to make the cognitive links between the two.

The new revision will be usable several ways. Students will still be able to use images from their own videotaped motion events. The new version will also support access to videodisc images. And it will be possible for students to analyze "artificial" motion events produced either through programming, utilizing a painting program, or by using a simulation environment like Interactive Physics. So, besides its obvious use as a data gathering and analysis tool, *VideoGraph 2.0* will also be used by students to analyze previously recorded motion events or even simulated microworlds where the laws of motion are programmed into the system by either themselves or their teacher.

It is important to realize that my purpose is not to eliminate labs or replace them with simulations. I want to establish a student tool which can help in the study of lab phenomena and real world events, and yet still be used (either as homework or in class) to analyze previously recorded or artificially produced events. In other words, the same software package would serve as a general motion data collection and analysis tool and as a simulation for further study. I feel a successful first step toward this goal was made with the earlier version. These materials are currently in use in schools in the U.S. and the U.K. The project has been favorably reviewed by the teaching community and, in fact, won the 1989 Ohaus Award for Innovation in Science Teaching from the National Science Teachers Association.

A new kinematics graphing test, currently being examined for reliability and validity, will be used to evaluate the effectiveness of the materials. Additional assessments, including participant observations and interviews, will provide a more detailed picture of student learning.

This research was supported, in part, by the National Science Foundation (MDR-9154127). Opinions expressed are those of the author and not necessarily those of the Foundation. Additional support came from RasterOps Corporation, SONY of America, and Apple Computer.

ERIC

HyperGuide: An interactive student guide for a virtual classroom

GILLES BERGERON
GILBERT PAQUETTE
JACQUELINE BOURDEAU
LICEF, Télé-université,
1001 Sherbrooke Est, 4è étage,
Montréal (Québec) H2X 3M4

The LICEF (Laboratoire d'Informatique Cognitive et Environnements de Formation) tackles the research and development problems of integrating new technologies to learning environments. In open universities, one of the solutions is to re-create the dynamics of a university campus in the form of a **virtual classroom**. One way to achieve this is through a HyperGuide, an environment that combines three new technologies: knowledge-based systems, hypermedia, and telecommunications. It was first intended to serve as a substitute for the "student guide" that is traditionnally sent to the distance education student, to guide her through the lessons of a course.
Developed from a constructionist perspective, it helps the learner find her own way through the various learning activities, and build her knowledge from them. These activities are the nodes of a partially-ordered network: from a node, the learner can choose from several paths, some choices being limited by such constraints as the successive steps of a method, or pre-requisites.
The learner will find the following in the HyperGuide:

- A general description of the course, its objectives, contents, and evaluation modalities
- A description of the learning activities, in the form of instructions guiding the learning process
- A description of each document to be found in the document base
- A graphic representation of the structure of the different modules of activities, and of the course contents. Through this interactive representation, the learner has direct access to the activities: read a printed document, view a video document, use a software application, take part in a computer-assisted conference, etc.
- Advising tools that help the learner in her productions and through her learning process. Tips are based on a model of the knowledge required to successfully do the tasks
- Exercises and tests that are corrected on-line, so that the learner can assess her own progress
- Self-assessment tools based on data collected while the learner interacts with HyperGuide. The learner can use three tracing tools:
  - a recording of the activity she has just completed;
  - a table of events;
  - statistics on the sequence of activities, the time spent on each, and the test results.

Research questions

Designing the HyperGuide environment represents one step towards the architecture of distributed, knowledge-based learning environments. However, much research is yet to be done and many questions, among which the following, are yet to be explored:

- the cognitive modeling of the multiple dimensions of a virtual classroom
- the role of the professor
- the minimal systems requirements for integrating the three technologies within a single system
- the applicability of the approach to campus-based courses
- the relevance of designing a HyperGuide environment for the tutor, and
- the building of intelligent advising capabilities.

HYPERCASE - A Computer Software for Planning and Developing Hypermedia Systems

PAULO MARCONDES CARVALHO JUNIOR
Center for Biomedical Informatics (NIB)
State University of Campinas, PO BOX 6005, Campinas, SP, Brazil, 13081-970

Hypermedia projects require an intensive stage of planning, as do ordinary software projects. One major difficulty is the need to gather in the same team various persons with different professional profiles, such as software engineers, designers, video makers, educators and subject experts. These persons need to exchange information, document their projects and demonstrate their ideas to each other. CASE (Computer-Assisted Software Engineering) tools help software engineers and data processing personnel to specify and plan software. Their basic function is to offer a project structure and a set of formalisms and rules on which to base software development.

Analyzing this situation, we have defined our work, regarding the evolution of CASE, in an almost opposite way. That is, while CASE appeared as a late evolution of structured systems methodology, we have chosen, on the contrary, to first produce tools suitable for hypermedia design, and then, in a subsequent phase, if the methodology in use proves to be improper, to develop a new one, better suited to hypermedia work.

With these considerations in mind, we are implementing a basic set of hypermedia systems project tools: HYPERCASE, or CASE for Hypermedia.

The basic idea of our work is the manipulation of objects, in a way similar to objects found in object-oriented programming, but without an exact correspondence, since the concept of object hierarchy is lacking. Examples of objects would be still frames, sound, movies, etc. Each object has properties specific to its kind. Screens are the basic objects in the system; any other object can belong to one or more screens. For each object, there is a record with information about it, that can be accessed from any system module. Each record can have up to ten keywords, allowing many possibilities of logical indexing, automatically.

The purpose of one of the tools is to help create graphic representations of the project's specifications. Graphic schemes can be created by assigning symbols to various objects, generating navigation maps automatically. The tool checks for consistency, verifying the system's degree of completeness, its granularity, its infinitude and its interruptable character. One of the tool's modules is specifically intended to sort video frames into a map of "disc geography" for the generation of laser video discs.

We consider the existence of two kinds of links: physical and logical links. The former are originated from the natural chaining of the data; for example, a sequence of gastrointestinal X-ray plates. Logical links, on the other hand, occur as a result of a logic judgment or reasoning which puts together two or more initially disconnected objects. Thus, in our example, the record of a given X-ray plate can have the keyword "peptic ulcer"; if a movie showing a stressed man has the same keyword, these two objects can be linked to each other by a logical link.

The most important point in this line of work (and in some other related ones) is the creation of a basic framework to make the design and documentation of hypermedia projects easier.

Electronic Portfolios: A Multimedia Application

GEORGE A. COLLIER, JR., Ph.D.
Department of Accounting, Economics, and Finance
MARGARET S. KIRBY, Ed.D.
Department of Office Administration and Business Education
Southeastern Oklahoma State University, Durant, Oklahoma 74701, USA

Multimedia, hypermedia, and assessment are contemporary educational catchwords. They represent a growing technology and a growing mandate for educational accountability. They are applicable to all educational levels.

Accrediting agencies are beginning to enforce stringent assessment standards. Agencies are Federally mandated to "determine that institutions or programs document the educational achievements of their students ..... in verifiable and consistent ways" (U.S. Government, 1991, 34CFR 602.17). Portfolio review is one suggested way.

An artist is almost expected to present a portfolio of work. Portfolios can be similarly understood as a device for collecting student assessment data. However, the image of a file containing samples of a student's work conjures up a corresponding image of voluminous amounts of unwieldy, easily misplaced hard copy.

This demonstration explores multimedia/hypermedia as a platform for storing, evaluating, and presenting portfolio information. A constrained budget limits hardware to information captured on video tape and floppy disk. Scanned input and erasable optical disks are other logical options as they become cost-effective.

This project is not only an application and evaluation of extant technology to portfolio assessment but also investigates rubrics for consistently evaluating assessment information captured. The adage *beauty is in the eyes of the beholder* is an evaluative phrase applicable to an artist's portfolio. However, accreditation agencies will not accept this method as "verifiable and consistent" documentation. In an era of outcomes based education (OBE), assessment requires that educators prove that their processes add value to their products. Moreover, assessment requires not only demonstration of value added from process input to output but also longitudinally (throughout the lifetime of the student). In short, educational accountability is demanding more hard evidence than assessment tools presently appear to satisfactorily muster.

Asymetrix's *Toolbox* authoring language and a NEC *PC-VCR* are among the tools used for crafting this approach. Accomplishments of students in the 1993 Business Education methods course at Southeastern Oklahoma State University were selected as the content for the electronic portfolios. The students and the instructor participated in the identification of criteria for selection and criteria for judging merit. Teaching demonstrations, lesson plans, test scores, and writing samples are among the items included in the students' portfolios.

The researchers envision the expansion of this primitive electronic portfolio paradigm to a more sophisticated collection of students' assessment data throughout their collegiate career and afterward. The researchers hope the demonstration serves as a focal point for educators interested in portfolio approach to assessment. Input, fresh ideas, and discussion are welcome.

Multimedia, Constructivism, and Higher-Order Thinking Skills

CHRIS DEDE, LYNN FONTANA, AND CHARLES WHITE
Center for Interactive Educational Technology
George Mason University, Fairfax, VA 22030, USA

Unless information tools that aid in teaching thinking skills are embedded, multimedia technologies will further overload teachers and students with masses of intellectually indigestible data. Our research is yielding insights into multimedia design principles to stimulate learner cognition and motivation. We are constructing a software shell that inculcates inquiry skills while helping students acquire content knowledge via multimedia materials. The shell is being applied to a history prototype based on Ken Burn's award-winning PBS series, *The Civil War.*

The Multimedia and Thinking Skills Project proceeds from the premise that the most appropriate role for multimedia in schools is not to augment data delivery in conventional instruction, but instead to foster a new model of teaching/learning based on learners' navigation and creation of knowledge webs through a formal inquiry process. Higher-order thinking skills for structured inquiry are best acquired under conditions of:

* active construction of knowledge by students rather than passive ingestion of information;
* the use of sophisticated information-gathering tools that allow pupils to focus on testing hypotheses rather than on plotting data;
* the use of multiple representations for knowledge, so that content can be tailored to individual learning styles;
* collaborative interaction with peers, like team-based approaches in modern work settings;
* individualized instruction that targets intervention to each learner's current difficulties; and
* evaluation systems that measure complex, higher-order skills rather than recall of facts.

The prototype software shell has four generic, essential features: the IBI (Inquiry Bureau of Investigation), Dr. Know, the Production Console, and Custom Tours. The IBI provides explicit instruction in inquiry, while Dr. Know is a context-sensitive coach who helps learners develop data-gathering and metacognitive skills. The Production Console gives students tools with which to manipulate the information in the database and create their own tours. Custom Tours is a non-structured access system to the data that allows free-form searches based on students' interests. By incorporating these features with extensive multimedia databases, the prototype demonstrates:

* explicit instruction in the steps of inquiry in the IBI;
* implicit instruction in the steps of inquiry and in modeling thinking strategies and knowledge structures via Guided Tours;
* a context-sensitive coach (Dr. Know) who helps students use their skills as they encounter the knowledge base;
* a student/teacher controlled production/editing function as a vital element in both instruction and evaluation;
* an approach to video as an integral part of a rich database to view, analyze, motivate, and evaluate; and
* large bodies of primary source materials that are the building blocks of the knowledge base.

Design work on this project was conducted under funding from the Corporation for Public Broadcasting and from George Mason University.

A Hypertext Exploration of Visual Perception

KEVIN M. FABER AND JAMES J. DEHAVEN
Division Of Mathematics and Natural Sciences
D'Youville College, Buffalo, NY 14201

The scientific education of students in the health sciences is critical to their later development as doctors, nurses, physical therapists, and as practitioners in scores of other allied professions. In their pre-professional courses, and throughout their careers, these men and women will be repeatedly confronted by situations which require them to analyze information from many disparate areas of science and mathematics.

Students entering these fields frequently derive their attraction to the health sciences from a sense of personal idealism, or, perhaps, from of a fascination with human biology. They bring with them the well-documented problems of inadequate preparation in mathematics and the physical sciences, problems which may be compounded for those returning to school from the work-force. Students in this position are frequently confused by casual references, in their course work or in journals, to topics derived from physics, chemistry, molecular biology or mathematics.

Hypertext and hypermedia are particularly suited to address difficulties of this nature. They can, in principle, provide the perfect textbook, in which the learner can access immediately an explanation for questions in areas of knowledge which might be ignored or relegated to an appendix in a conventional textbook. Our implementation of this ideal consists of a series of modularized units, each of which addresses a relevant scientific topic, with the appropriate computerized links between each such subunit:

1) The topic, human visual perception, illustrates the kind of multidisciplinary problem which is particularly amenable to this kind of treatment;
2) Equal weight is given to all relevant scientific areas;
3) Modules are targeted at the scientifically literate person or at the student motivated to study the topic;
4) The modules are interchangeable and a front-end is provided for connecting them.

An understanding of human visual perception requires, at the minimum, some degree of literacy in cellular anatomy, neuroanatomy, electricity and magnetism, wave optics and geometric optics, chemistry and biochemistry, as well as the psychology of perception. None of these topics is peripheral to our understanding of the whole and so each must be treated on an equal footing with all the others. Our program tackles the problem frontally by incorporating sections in, for example, chemistry and physics which are as richly developed as those in biology

The program, developed using Claris' HyperCard and MacroMedia's Director, is provided with a front end for assembling and linking additional units. It is richly supplied with animations and graphics. For example, one animation illustrates the response of the visual cortex to a rotating line-segment; another illustrates the interrelation between the frequency wavelength and color of light. The application also contains a concise on-line navigation system. Some useful alterations in mouse handling routines which might be of interest to other developers have also been made.

ERIC

Chronoscope: Teaching History with Interactive Media

ANNE J. FAVALORO
Producer, Interactive Projects
WGBH Educational Foundation
125 Western Avenue, Boston MA 02134

WGBH Educational Foundation in Boston is perhaps best known for its national productions that have aired on the Public Broadcasting Service.

In addition to its broadcast programming, WGBH develops products and services in new media technologies. One of WGBH's recent products is *Chronoscope*, an innovative videodisc series for middle and high school social studies and history classes.

Exploring 1492 is the first videodisc title in the *Chronoscope* series. Students use this disc to unravel the mysteries of Columbus and the Age of Exploration by exploring aspects of life in the 15th century.

Throughout the videodisc, students view short, video introductions, supplemented by detailed text, graphics, motion-video clips, and still images. The material is organized into a curriculum focused on world cultures and encounters that existed during the age of European exploration.

An Explorations section depicts the world of 1492 through geography, daily life, trade, and civilization in each of five areas: the Americas, China, West Africa, the Arab World, and Mediterranean Europe. In each area, students glimpse life through the first-person account of someone living in 1492.

A Stories section provides short documentaries related to the encounters and legacies of the world in 1492. These topics provide more food for thought: Crossing the Seas; An Exchange of Plants and Animals; Who Discovered America?; Slavery in the Americas; After 1492 - New American Cultures; and Should We Celebrate Columbus?. In the Activity section, students use a computer game to browse and do research within the program. And an easy-to-use software Editor lets teachers and students combine their own text and graphics with materials from the videodisc and software to create customized presentations and reports.

Chronoscope: Exploring 1492 includes a two-sided videodisc with 60 minutes of motion video and numerous still images and maps. The software is for the Macintosh platform. A printed User's Guide and a Teacher's Guide, with videodisc bar codes and student worksheets, are also provided. The product is published and distributed by Optical Data Corporation.

Exploring 1492 is designed to be used in conjunction with textbooks and other teaching materials. It captures students' interest by bringing history to life in ways that textbooks cannot. By providing awareness of many different perspectives on the era, *Exploring 1492* allows students to look at the world of the fifteenth century, and perhaps to see their own world in a new light.

In addition to *Chronoscope*, WGBH interactive productions have included three titles in the *Interactive NOVA* videodisc series, published by Scholastic, Inc., for middle/high school science classes. WGBH titles currently in production include: *Interactive NOVA: Earth*, a videodisc program aimed at middle school earth science classes; an interactive biology videodisc for the high school/college level; and an interactive companion to a new high school U.S. history textbook.

Developing Competence of Utilizing Informat.on by Using Authoring Software for Creating Computer Games

MAKIO FUKUDA SHINJI KIMURA KATUAKI KURODA

Department of Human Science *Department of Home Economics*

Osaka International University for Women. Moriguchi, Osaka, Japan, 570.

KANJI WATANABE

Osaka Electro-Communication College

Neyagawa, Osaka, Japan, 572.

OSAMU IEMOTO

Baika College

Ibaraki, Osaka, Japan, 576.

In everyday life, we face many situations in which we must solve problems and make decisions, and when doing so, we often need to gather and analyze all kinds of information. One's decision is usually made on the basis of the information processed. Our research has focused on the analysis of the methodology to develop the competency necessary for problem solving. In order to devise an effective method, we first outlined the seven concepts (see Table 1) which constitute the competence to interpret and apply information successfully. For the activity with which the learners can develop the concepts, we assigned our students to create computer simulation games. In the process of making complex scenarios, the gamemakers need to go through the stages which increase their awareness of various types of information and how they interact each other. Table 1 shows the relationship between the concepts defined and the criteria for the evaluation of games.

Table 1: The concepts and the criteria for evaluation

The Criteria for Evaluation	The Seven Concepts
creative design	motivation / attitude
objectives of the game	aims for information utilization
game materials (graphic/text)	consideration on the quality of the information
game content, format and structure	elaboration of the information
players' behavior	behavior caused by the information
after game feedback	evaluation of the result of the information utilization
morality	consciousness of norm and morality

The students of our university (the learners) made some interesting multi-media computer games. The questionnaire administrated at the end of the course indicates both that there was considerable amount of effect of the activity on developing the competence of utilizing information, and that the students had high motivation and positive attitude towards the activity.

COMPUTER LITERACY FOR THE 90'S: A Multienvironment Multimedia Event!

Susan A. Garrett
and
Dolores Pusins
Department of Computer Science
Hillsborough Community College at Tampa, FL 33630

Focus: The redesign of the Introduction/Literacy course being taught on higher education campuses.

Purpose: To design and present a model for incorporating multimedia technology into the presentation of information in beginning level computer information courses. Information that has, historically, been presented through lecture, demonstration, use of overheads and other visual aids will now be presented in a much more dynamic, high-tech manor enhanced by hands-on activities in lab settings.

Tools: Since this is designed to span both the Macintosh based environment and the DOS based environment, we will use both types of equipment, and software(such as Macromind Director and Windows Player) for the presentation.

Summary: Multimedia incorporates:
-text
-graphics
-animation
-audio
-video
and can be a most effective tool for enhancing the learning environment in that these techniques bring the content to life more powerfully than other types of presentation software or, certainly, more so than traditional techniques. Simulation and prototypes can provide visualization of complex processes or concepts with interactive capabilities allowing for concrete learning. Presentations, once created, may be recorded onto videotape to be distributed inexpensively to other users.

It is our intent to develop a number of module(s) to assist in presenting information to students enrolled in the introductory courses in such a manner as to enrich their learning, as well as enhance our marketing efforts to increase program enrollment. We feel that the excitement generated through well-designed multimedia presentations will help us achieve those goals.

Exploring Mathematics with Interactive Texts using MathKit

LADNOR GEISSINGER
Inst. for Academic Technology, UNC-Chapel Hill, Box 12017, RTP, NC 27709

Interactive math texts are computer documents that may contain text, calculations, graphics, symbolic expressions, animations, video clips, and code that can be run as the user interacts with the text. Maple, Mathcad, and Mathematica provide for electronic notebooks which are basically scrollable single-page documents that expand or contract to accommodate user editing and program output. Typically the user enters commands and the results are displayed in space automatically opened at the cursor position, often scrolling earlier results off the screen. These supercalculators have powerful and somewhat complicated command-line languages designed for use by engineers, scientists, and professors, but few students master enough of the detailed instructions to experiment with more than simple modifications of their instructor's notebook.

By contrast, the interactive math texts produced with MathKit are more like books; they are paginated hypertext documents. The author designs the book as he/she wishes, arranging on each page any combination of fixed background or scrollable text fields, command lines, graphing windows, direct manipulation gadgets, or even a complete programming setting. The author decides on the type of learning environment that seems most appropriate for the students and the particular math topic. It can be narrowly focused and require little previous experience, or it can be set up for free exploration of a topic with a lot of student input expected. Regardless of the style of the book, or the initial setup, the reader has available a menu of easy-to-use exploratory tools to activate at any time. We hope thereby to encourage play and experimentation.

MathKit was designed from the start as a tool for learning math. It is intended to facilitate the creation and use of interactive mathematics books at the high school and college level. It is being developed at the Inst. for Academic Technology, a partnership between the University of North Carolina at Chapel Hill and IBM's Academic Information Systems. Its chief architect is math professor Jim White. MathKit extends the Microsoft Windows environment to make it easy to handle mathematics -- to enter formulas and programs, to carry out symbolic and numeric calculations, to display and manipulate the output, and to read and write mathematical texts. The brains of the system is a LISP interpreter.

MathKit is a library of tools which can be used with various front end programs such as ToolBook, Visual Basic, and the special authoring program we are building, called TIM (Toolkit for Interactive Mathematics), which was designed with the help of high school teacher Jim Swift. An author can lay out a page using the built-in MathKit display windows (text, math edit, data, graph2d, graph3d, animation, program) and user gadgets (sliders, buttons, check boxes). The author writes short scripts to activate the display objects and gadgets, and so constructs a math microworld. There is an extensive command and programming language called MathScript, which is English-like, informal with a forgiving syntax, intuitive, and very easy to learn.

ToolBook (a software construction kit for Windows) is a widely used presentation tool. By itself it is inadequate for building interactive math books, but in conjunction with MathKit it is a great tool. TIM has less variety in objects and settable properties than ToolBook, but it is also much easier to learn to use and it has two special explorer pages: for algebra/functions/calculus and data analysis. To do some real data collection and analysis in class, connect your computer to an IBM Personal Science Laboratory unit, plug in a probe or two, and import data directly into your interactive math book. These are the instruments for mathematics as a laboratory science.

A Hypertext Dissection Atlas

D. EDWARD HART, KEVIN M. FABER AND JAMES J. DEHAVEN
Division of Mathematics and Natural Sciences
D'Youville College, Buffalo, NY 14201

The student in the introductory course in gross human anatomy is confronted with a daunting task. A huge volume of information must be committed to memory, much of it consisting of an arcane and obscure nomenclature. In addition, the structural and functional relationships among the diverse components of the body's subsystems must be understood and recognized.

Traditionally these objectives have been accomplished by means of a lab experience in which the student dissects a human cadaver. Although the experience acquired through this method is irreplaceable, there are several practical problems which inhere in this approach:

1)The number of human specimens is limited, and students rely on presentations by other students for a large portion of their experience;

2) The cadavers deteriorate during the course of the laboratory;

3) The crude dissecting technique of beginning students frequently destroys details of innervation and blood supply for subsequent observers;

4)There is considerable variation among specimens which, although it can illuminate anomalies, can also confuse the beginner.

It is therefore reasonable to explore ways in which computer-assisted-learning can be used to augment the traditional lab experience. Because human anatomy requires reference to a large and complex database, and because there is a concurrent need on the part of the learner for experimental visualization of the relations between members of the database, there has been considerable interest in both commercial and non-commercial utilizations of this technology. Because of our belief in the paramount importance of the lab experience, we have developed the first module of a hypertext program to be used in conjunction with our gross human anatomy laboratory course. The program, developed using Claris' HyperCard and MacroMedia's Director, consists of a HyperCard stack which we hope will serve as an animated "atlas" for our students.

The current work, although limited to the dissection of the back and the upper arm, already contains hundreds of digitized photographs of dissections and of models, as well as QuickTime movies of dissection techniques. Animated displays of blood supply, innervation, origins and insertions of muscles, and the position of partially occluded structural details are superimposed upon photographs of dissected cadavers.

Because it will be tool to be used by laboratory students, the central organizational principle of this application is the routine of the conventional gross anatomy lab. Navigation through the stack is according to anatomical subsystem and to the dissection layer of that subsystem. The application also contains practice quizzes similar to our written exams as well as visual tests similar to our lab practicals. Current work involves automating the processes of adding graphics and inserting additional subunits into the stack.

"The Use of Multiple Technology Resources in a Distance Learning Doctoral Program"

Lois Ann Hesser
Al P. Mizell

The National Cluster in the Child and Youth Studies doctoral program at Nova University is described through a series of posters, videotape, and handouts to provide an overview of what is involved in developing and delivering a doctoral program via telecommunications using a variety of media resources. Nova began delivery of distance education in 1972 and introduced online computer delivery in 1983.

Distance education is defined as instruction that occurs at a distance from the instructor (Feasley, 1982); or the delivery of instruction in which student learning takes place at one location, while content and management expertise are at another location (Evans, 1986). Distance education that involves the use of a personal computer and modem is becoming more popular today to electronically deliver instruction and to provide real-time interaction between the faculty member and a group of students.

The Nova Child and Youth Studies Doctoral Program offers its degree programs using an electronic format, known as the "National Cluster." This online approach is offered as an optional alternative to the more traditional weekend cluster format where the professor is flown into a regional site to meet with the students in a traditional class setting. Instruction is accomplished through a combination of "live" meetings, electronic communication, and various audiovisual resources. Among the online activities to be described are the use of electronic mail and "real time" participation in electronic classroom (ECR) sessions. A description of the way the program is delivered and how students react, as well as examples of the materials, videotapes, study guides, phone contacts, etc. that are provided will be shared.

Lo v Cost Multimedia In Teaching

GILLES KUNTZ
Institut IMAG - Grenoble University
B.' 53X F-38041 GRENOBLE Cedex (France)
E-mail: kuntz@imag.fr

Today the availability, at an affordable price, of graphics cards, the progress in the techniques of still or animated image compression, the integration of these tools in QuickTime enable the realisation of multimedia presentations even in fields with low financial possibilities such as education and professional training. The aim of this poster is to show that the financial and technical obstacle have disapeared. Therefore one should not hesitate to implement without delay these new technologies augmenting this way, without any doubt, the power of communicating.

Developing Multimedia Applications

What are the options possible today to a multimedia developer ?
Two classes of developing tools are available :
- authoring systems : although these softwares pretend to be acessible to non computer scientists, the truth is that their use needs logical concepts that are close to the formalism of programming; therefore there are not easily usable by the non specialist and the resort to computer scientist is still necessary.
- prototyping environments like HyperCard : at the opposite of the latter these systems do not pretend to avoid the use of a computer scientist for the realisation of the multimedia application prototype specifics scripts , use of external commands...), but their flexibility enables the end user to modify himself the presentation in its final aspect (choose of images, of sounds, adjonction or deletion of screen pages which act as a model page, change of texts...). This very important problem of update in multimedia is very often forbidden because of the complexity of the tools and techniques used.

Color HyperCard External Commands

These tools developed in IMAG Institute are a set of external commands for HyperCard version 2.0 and after named 'Color' in the purpose of adding colors to HC. These external commands deal with a special color window on top of the HyperCard window. Mouse events in the color window are sent to HC card, so all the user interaction is under control of HyperTalk scripts. Special XCMD can draw styled text extracted from text fields over the color window and others let the user to update texts in field in a convenient manner. These kinds of tools allow an end user to modify in a simple way texts, images or perhaps buttons accordingly to scripts written by the developer.
With this set of XCMD a complete interactive application with buttons, text fields, sounds, pictures can be prepared in a short time with the help and the power of HyperCard.

A complete demonstration of these tools can be found by file transfer procotol on INTERNET at imag site by anonymous ftp to ftp.imag.fr in the /pub/COLOR-XCMD directory under filename color-xcmd.bin.hqx

Creative Interactive Programs for the Middle School Student

SAMUEL A. LAPPANO
Catherine Hall Middle School
Milton Hershey School
Hershey, Pennsylvania

Current interactive software offerings seem to address the educational needs of the elementary and secondary students. There seems to be a lack of good interactive software on the middle school level. Many programs on the middle school level seem either too elementary or too advanced for thi s unique educational level. I have written and developed HyeprCard programs in the following three subject areas for use in our middle school program; History, Music and Geometry.

The first program to be demonstrated is "The Battle of Gettysburg." The program uses the uniqueness of the HyperCard format to present information regarding this historic Civil War Battle in an innovative format. Students are free to explore text and graphics concerning the battle. As they read the narrative of the battle, they may freely move to maps of the battle as well as click and print any of the battle maps.

An animation sequence of cards shows the movements of the Union and Confederate troops. At any point in the program, students may also explore a pictorial file of the generals involved in the battle. Each file contains a scanned picture of the general and also biographical information. Many histc ic photographs have been incorporated into the program. The value and uniqueness of the program can be appreciated through a demonstration of the program with audience participation.

The Second Part of the demonstration is in the area of music education; the stacks are entitled, "The Magic of Mozart" and "Beethoven and His Music." The programs feature extensive sound resources as both instructional parts of the program as well as reinforcement tools for the review section. Unique features also include visual choices and reinforcements not possible if the information was presented only in text style. Once again, the demonstration of these programs as well as a visual presentation on how such elements can be implemented will be of great value to the educator.

The final demonstration will be a stack entitled, "Geometry - All Shapes and Sizes." The stack was developed through the John Lilly Grant Program. The object of the grant was to devise new and unique ways to convey mathematical principals to middle school age students. The program has an instructional section which uses a graphic method of demonstrating geometric terms for the user. Scanned photos and original drawings that incorporate the geometric shapes being discussed and reviewed are just one of the innovative methods used. Sound, graphics and text combine in an innovative manner to present geometric concepts for the middle school age student.

In addition to the original programs described above, several commercial HyperCard programs will be domonstrated and discussed.

DESIGN, DEVELOPMENT, IMPLEMENTATION AND EVALUATION OF COMPUTER-BASED INSTRUCTION (CBI) FOR SOME TOPICS OF GENERAL CHEMISTRY COURSES.

Manuel Martínez M.; Santiago Fernández S.; Mónica Herrera C.; Yolanda Plá F.
Departamento de Química; Facultad de Ciencia Universidad de Santiago de Chile. Avda. B.O'Higgins 3363; Santiago de Chile

The adoption of new technologies aimed to improve the education quality and to encourage the participation of students, is something widely known. From this point of view, the rapid proliferation of relatively inexpensive microcomputers, added up to a more sophisticated technology, and a significative amount of commercially produced educational software programs, has brought a notorious increase in the use of them in educational institutions. Surprisingly, research studies about the educational effectiveness of individual software programs are rather few as it was pointed out by Jolicoeur and Berger (1986). Plom and Pelgrum (1991) concluded that computers are used mainly for teaching about computers. Niemiec and Walberg (1987) after comparing and evaluating 16 reviews, concluded that what is known about CAI is that is a moderately effective instructional intervention. Whiting (1984) points out that computer assisted learning (CAL) has become a powerful instructional and learning medium, but the concept of "mastery learning" has not been applied to CAL strategies.

The analysis in the specific case of use of computers in the area of chemical education, allows similar conclusions. Software available to be used in tutorials or self-instructional materials, is scarce, if any. As a conclusion, it can be said that the available results deserve further research in this area.

In our university, we have been working in the design, development, implementation and evaluation of educational software to be used in chemical education, that is effective, efficient, and encourages the participation and interest of students. The subjects were prospective chemistry and biochemistry students attending General Chemistry courses . The instructional materials were systematically designed by using Dick and Carey's model (1978).

The results have been quite satisfactory. The most interesting situation that has derived from this research, is the data base that has been gathered about common mistakes that allowed us to design computer dialogs for teaching those topics that offer major problems to our students.

References.

Dick, W.; Carey. L.(1978). *The Systematic Design of Instruction*, Scott, Foresman and Co.,Illinois, 1978.

Jolicoeur,K.; Berger, D. (1986).Do we really know what makes educational software effective? A call for empirical research on effectiveness. *Educational Technology*, 26(12), 7-9.

Niemiec,R.; Walberg, H.(1987).Comparative effects of computer assisted instruction:A synthesis of reviews. *Journal of Educational Computing Research, 3*(1),19-27.

Plomp,T. & Pelgrum, W.(1991).Introduction of Computers in Education: State of the Art in Eight Countries. *Computers & Education, 17* (3), 248-258.

Whiting,J. (1984).Cognitive and student assessments of a CAL package designed for mastery learning. *Computers and Education, 8*(1),59-67.

ERIC

Instructional and Research Uses of Multimedia As Learning Empowerment for Faculty and Students: Teaching Chinese Art History with AVC

FRED H. MARTINSON
Department of Art
University of Tennessee at Knoxville, TN 37996, USA

The basic problem in Asian art history is not very different from other disciplines in the humanities. It is how to make available the materials for any given class, and, for more advanced students or for research projects, how to make available the larger resources behind any given class taught. Art history does require the use of many images, but the image capablities of the AVC program are outstanding.

I am teaching Asian art history almost entirely with the use of the images, text, and programs generated by the Audio Visual Connection (AVC) multimedia program made by IBM. This multimedia project (now called the "AVC Project") at the University of Tennessee/Knoxville (UTK) is based in the College of Education (COE) in the Department of Educational Leadership, and it serves the Asian art history area of the Department of Art in the College of Liberal Arts. At present it is used primarily for classroom teaching, but once computer laboratories are expanded, students will be able to review and work ahead using the same images and information as the instructor. This rather different history is related in brief below. Much of the audiovisual data already gathered will go toward creating a visual, research database and a tailored curriculum. With new developments such as Digital Video Interactive (DVI), the training of the teaching and support staff and interested students is an on-going process.

For ED-MEDIA 93, I propose to demonstrate the present capabilities of this project by allowing the viewer to use our present trigger fields to navigate to any Asian culture and to any period in that culture's history of art. For students in surveys - even in the classroom, such a hypermedia capability gives them the power to review and/or work ahead; for faculty and advanced students, it means that detailed investigations can be located beside or within their appropriate historical and spatial places. Two aspects that are clearly better and very different from past methods in art history, are the potiential availability of all color images on servers (or eventually on laser disks or CD-ROM disks), and the ease of adding text information to class presentations. The difference is that although an intructor may use color slides in traditional teaching, students still have textbooks with mainly black and white illustrations. Also, it should be noted that programs such as AVC allow the calling of random access slide projectors and PC-VCR's, both traditionally used as quite separate devices in art history classrooms. I have found that text information in computer presentations replaces most, but not all, needs for the chalkboard (art historians do not usually use "text" slides, nor do they use overhead projectors; students rely on handouts and chalkboard information).

A Model for the 21st Century or Where are the Textbooks? They Have "Gone" Multimedia.

Ellen Nelson, C.E.O.

Decision Development Corporation
2680 Bishop Drive, Suite 122
San Ramon, CA 94583 USA

Today the printed textbook is the primary resource used in classrooms to define and present the curriculum to teachers and students. Materials supplemental to the textbook such as other print, movies, kits, video tapes, laserdiscs, computer software and integrated learning systems are also used. An increasingly dense base of technology exists in the classroom such as computers, videodisc players and CD drives to the degree that it now could be used as a comprehensive curriculum delivery system with or without textbook support.

Science 2000®, developed under a technology grant from the California State Department of Education, is a major first step in the transition from a primarily print to a technology delivery system. Based upon the California Science Framework and a constructivist pedagogy, Science 2000® provides teachers and students with a hands-on technology-based curriculum. Data bases of information, live action video, still pictures, and scientific tools, which allow simulated experiments, data collection and analysis, are the resources used by students to answer investigation questions and to explore and learn science based upon the questions and their own interests. The delivery platform for Science 2000® is an individual DOS or MAC personal computer or a network of computers supplemented by videodisc or digital hardware in classrooms and schools.

The applications of the future, or the next generation of Science 2000®, will be a multimedia curriculum delivered on-line. Fiber optic networks already in place, satellite telecommunications, and cable television networks throughout the nation will provide the means for accessing vast resources of information. The classroom as we have known it could change dramatically. Students will be able to access the curriculum and other information anywhere there is a computer, a phone or cable outlet. Completing lessons, receiving assignments, gathering information, solving problems, turning in homework and all the other activities that now take place in the school could take place anywhere. The structure of curriculum essentials can be described, sourced and delivered from schools, but information and resources will be sourced from a variety of loci such as the U.S. Weather Service. In the 21st Century, the textbook as the primary definer of the curriculum will disappear, replaced by a plethora of resources, by commonly agreed on essentials, and by as yet unknown technologies!

ERIC

Creating Interactive Video Units That Span

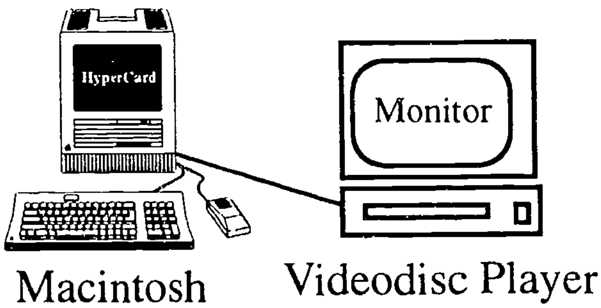
the Curriculum Using HyperCard and a Laserdisc Player

GERALD POLLARD
Interactive Video Instructor
Texas School for the Deaf - Austin, Texas 78764

The videodisc is visually a very powerful presentation tool. When combined with computer assisted instruction, interactive video provides the most powerful learning environment affordable today. This demonstration will illustrate how to create instructional interactive video units using a Pioneer videodisc player, a Macintosh Computer, HyperCard and The Voyager Videodisc driver.

A description of the two classes taught (HyperCard and Interactive Video and Advanced Interactive Video) will be provided. Examples of student produced interactive video units will be demonstrated as follows: for elementary science, Optical Data's *Windows on Science* videodisc will be used. For whole language the videodisc *Sarah Plain and Tall* will be shown. For photography the *Visual Almanac* videodisc will be shown. For English sentence patterning *Indiana Jones and the Last Crusade* videodisc will be shown.

A handout regarding interactive video information and references will be provided.



Audio Dictation, a Talking Software to Teach Language Art

Minerve M. Robert
Adult Education
Commission scolaire des Mille-Iles
Laval (Quebec) Canada, H7L 2P1

Language Art, to be complete, has two dimensions: oral and written. Up until now, even our most sophisticated computers conveyed only the "visual" or written facet of learning. The apparition of affordable sound cards was a waited-for breakthrough allowing a combined audio and visual approach to the teaching of language.

The "Audio Dictations" is an open software that teaches and make computer talk. The software was developed specifically for the teaching of a language using a SOUND BLASTER card with MS-DOS technology. It was created by Dr. Claude Guillette, a professor at the "Université du Québec à Montréal" and Dr. René Beauchamp, a curriculum consultant at the Mille-Iles School Board in Laval. It is an empty "shell" that can be filled with appropriate language skill contents. It is also adaptable to all levels of students. Dealing with a mother tongue, the teacher inserts the words of vocabulary, according to the curriculum imposed by the educational authorities.

The teaching of a foreign language can fill in similar exercises to prepare the students to oral and written communication. The sound aspect adds to the wonders of interaction between the computer and the learner. Reinforcement or explanation immediately follow each word typed. As a bonus, let us add that this software allows a follow up possibility, a management of the results of students, individually and by group.

What about the impact of a talking computer versus the mute version? Having tried "Audio Dictation" with young adults, our students have proven that sound is a worthwhile addition: their interest is sustained, their concentration is heightened. Even their own elocution has improved because they are confronted with a correct pronunciation of each word, before they write an answer.

We have produced many French dictations containing an average of 20 words to type per dictation. We have also produced exercises containing more than 1 000 of the basic French words, presented in a list of words to ear and type correctly. The same software was also used to produce many different exercises to learn English as a second language (In Quebec, French is the mother tongue of 80% of the population). The software could be used to teach most modern languages. These programs are now available through DIDACTIK*, a Quebec firm.

We believe a talking computer can become an invaluable friend in learning languages. A word or a sentence can be repeated as many time as desired. A new dynamic is now available. Most phone companies are using talking computers, why not education?

----

* DIDACTIK, 162 André-Chénier, Laval, Quebec, Canada, H7L 2S6

ERIC

Integrated Multimedia in Distance Education

FARHAD SABA
Department of Educational Technology
San Diego State University, San Diego, CA 92182-0311

RICK L. SHEARER
Director, Research and Evaluation
National University, San Diego, CA 92108

A study was conducted to empirically verify three constructs of *transactional distance*, *dialog* and *structure* in the dynamic theory of distance education. The study simulated the dynamics of discourse between an instructor and a learner (N=30, each learner received the instruction one at a time) while they used a synchronous, computer-based distance education desktop workstation. The hypothesis of the study was that increased dialog in distance education decreases structure and transactional distance. As defined by Moore (1983, p. 171) *dialogue*, is "the extent to which, in any educational program, learner and educator are able to respond to each other; " and *structure* is "a measure of an educational programme's responsiveness to learners' individual needs." In this view, *transactional distance* is a function of the variance in dialogue and structure in relation to each other (Moore, 1983); therefore, "distance" in education is not determined by geographic proximity, but by the level and rate of dialogue and structure (Saba 1990). Instructional content for the study was on the subject of culture and was produced in HyperCard.

A prototype voice, video, data integrated workstation, designed in an earlier study, and refined in the past three years, was used to deliver instruction (Saba and Twitchell, 1988/89). The learner and the instructor, working in two different locations, could see each other via a closed loop video circuit, and talk to each other via an ordinary telephone. Timbuktu was used to provide screen sharing capability between the learner and the instructor whose Macintosh computers were on a local area network. Instructional transaction between the instructor and learner was video taped on three recorders.

System Dynamics, was used to conceptualize the dynamic model of distance education, based on Moore's theory of distance education (1983). System Dynamics allows objective observation of each system function in terms of its present level of performance and the rate in which this level is decreased or increased through time. STELLA is capable of plotting the performance of each system function and the system as a whole in specific time intervals in the future. Rates of two main system components (dialog and structure) were determined by analyzing the discourse between the instructor and the learners. Discourse analysis is a technique of coding speech acts based on specific categories, or "phenomenon," (Beach 1990) for understanding "coherence and sequential organization" (Levinson, 1983) of natural conversation. Then, STELLA was used to simulate the relationship between these rates to see if transactional distance was minimized when dialog increased and structure decreased. Several data based-runs supported the hypothesis of the study. A dynamic relationship between structure and dialog was observed. As one increased, the other decreased. Transactional distance was maximized, when dialog was minimized and structure maximized.

References

Beach, W (1990). Orienting the phenomenon. In J. A. Anderson, (Ed.), Communication yearbook (pp. 216-244). Newbury Park, CA: Sage Publications.

Levinson, S. C. (1983). Pragmatics. Cambridge, UK: Cambridge University Press.

Moore, M. G. (1983). The individual adult learner. In M. Tight (Ed.), Adult learning and education (pp. 153-168). London, UK: Croom Helm.

Roberts, N., Andersen, D. F., Deal, R. M., Garet, M. S., & Shaffer, W. A. (1983). Introduction to computer simulation: The System Dynamics approach. Reading, MA: Addison-Wesley.

Saba, F. , & Twitchell, D. (1988/89). Integrated services digital networks: How it can be used for distance education. Journal of Educational Technology Systems. 17 (1), 15-25.

Saba, F. (1990). Integrated telecommunication systems and instructional transaction. In M. G. Moore (Ed.), Contemporary issues in American distance education (pp. 344-352). Oxford, UK: Pergamon Press.

Multimedia Training Systems & Network Security in the Campus Learning Center

Interactive & Multimedia Units in Information Technology

Peter M. Schleppenbach and Albert Bodenhamer
The Learning Center, Instructional Computing Lab
Los Angeles Pierce College, Woodland Hills, CA 91371, USA

Introduction

The charter of The Learning Center (TLC) at Los Angeles Pierce College (LAPC) is to provide quality support for all instructional programs, with priority given to learning and study skills for a diverse student population using current methodologies and technologies. Resources include one-on-one and group tutoring, a variety of audio-visual materials, and computer applications from word processing to basic skills for mathematics, English and English as a Second Language to advanced courseware for transfer and vocational education. Knowledge of information technology (IT) is an important literacy for transfer and general education. Access to this knowledge is equally important. In fact, the College is presently seeking a way to provide broad access to IT literacy so that this skill can be incorporated as a new graduation requirement. This project will produce a flexible and portable platform for independent study in information technology, to be pilot tested at LAPC.

The project consists of a series of custom and off-the-shelf applications tied together to perform point-of-entry support for students using the lab. When a student enters the lab, a card reader at the reception counter will begin to track the students access to resources. If it is the student's first visit to the lab, it will be noted on the reception computer and the student will be directed to take the Multimedia Tour of TLC. The Multimedia Tour of TLC will register him/her and then proceed to inform him/her of the resources available and how to access these resources. It will direct the student to a variety of other interactive and multimedia applications based upon his/her schedule of classes, major and his/her answers to questions during the session. It will also test the student for requisite skills necessary to access the applications using formative evaluation methods developed at UCLA and issue a TLC diploma.

Specific Application

Using Toolbook for Windows on an IBM 57SLC and Hypercard on a Macintosh IIci (object-oriented presentation applications) as development environments, the Multimedia Tour of TLC will be a demonstration of the potential for standalone training platforms in learning centers of higher education.

A Demonstration of the Creation of Computer-Assisted Interactive Radiology Courseware Using a Commodore Amiga

BARBARA SELCER
Department of Anatomy and Radiology
The University of Georgia-Athens, GA 30602, USA

Proficiency in diagnostic radiology is recognized as a complex skill involving the integration of knowledge of anatomy, pathophysiology, patient specific information, and visual skills for pattern recognition. The traditional approach to teaching veterinary radiology has been through lecture sessions with 35 mm slides supplemented by teaching files and textbook assignments. Since radiology is a visual discipline, active student effort is required in the learning process. The traditional teaching methodology is less efficient than interactive methods since it creates a passive learning situation for students. Computer-assisted instruction can create a more active student learning environment while allowing students to work at their own pace with the computer-assisted tutorial serving as a guide. Computer-assisted interactive radiology tutorials have been developed at the University of Georgia-College of Veterinary Medicine. These tutorials used the Commodore Amiga personal computer (Commodore Business Machines, Westchester, PA). Programs were authored using Amiga Vision (Commodore Electronics Limited and Insatt Corp, Westchester, PA). Tutorial screens are presented on a single monitor displaying both radiographic images as well as text. Students interact with the computer through the use of a mouse. Invisible "hit boxes" have been placed on the various graphical screens and become active areas when the students use the mouse. Throughout the tutorial, students receive instruction and feedback. Both audio and visual components are used throughout the tutorial for continued reinforcement of information.

The demonstration session will focus on 3 areas. The final program will be viewed. Following this there will be a demonstration of the authoring system incorporating computer generated animation, digitized radiographic images, artistically rendered graphics, digitized sound, and laser disc image incorporation. Unlike other platforms, all images are visualized on a single monitor and incorporated in a single program for interactivity. Finally, a brief demonstration of the authoring software and digitizing software will be presented. Actual radiographic images will be digitized and incorporated into a short mock tutorial.

Acknowledgements

The author gratefully acknowledges the financial support of the PEW Charitable Trust.

HyperBach: the *Goldberg Variations* and CD-ROM

TIMOTHY A. SMITH
School of Music
Ball State University, Muncie, Indiana 47306, USA

J. S. Bach's *Goldberg Variations* represent the greatest set of variations before Beethoven's *Diabelli.* The work was composed at the request of Count Von Keyserlingk, for his court harpsichordist Johann Gottlieb Goldberg. An insomniac, the Count desired that Goldberg should enliven his restless nights by playing what he affectionately referred to as "my variations." The *Goldberg Variations* were published as volume four of Bach's *Clavierübung*, or "Keyboard Practice." They represent a veritable catalogue of eighteenth-century keyboard styles, including idioms that anticipate virtuoso techniques of the nineteenth century. Each variation is based upon the same fundamental bass line, or *ruggiero*. This theme is stated most transparently in the opening *Aria,* a movement that first appeared in Anna Magdalena's notebook of 1725, where it was entitled *Sarabande.*

In addition to showcasing various keyboard styles, the *Goldberg Variations* also incorporate many important Baroque forms: dances, fugues, canons, toccata, ground-bass, and even a French overture. The overall form is *chiastic,* with analogous movements on either side of the central axis (variation 16). Variations that are evenly divisible by three are canons, with the quotient becoming the interval between leader and follower. The fourth and fifth canons are inverted. Three of the variations are in minor mode, and the three fugues are arranged symmetrically around the central movement.

Using an external CD-ROM disk drive the author accesses two compact disk recordings of the *Goldberg Variations* (Trevor Pinnock and Glenn Gould), synchronizing the sound with notation, animation and full-color graphics on the computer screen. In this presentation the author explores various interactive presentation techniques including the use of color, animation, layering of structural elements, aural-visual synthesis of events, and three dimensions. The program contains five ways of listening to, or studying, the work.

Just Listening: The user hears the music and has the option of following either a mini-score or program notes mounted on the computer screen.

Following the Score: The user hears the music while synchronized notation scrolls from from right to left on the computer monitor. Clicking the mouse in any measure will start the music in that measure. Clicking the mouse on an ornament will produce a pop-up window of realized notation of that ornament and the playing of that portion of the music from the compact disk.

Canon: The structures of the nine canons of the *Goldberg Variations* are presented with leader notes lighting up in red and follower notes in green. Leader and follower can be played separately on the Macintosh speaker or together from the compact disk.

Things to Notice: The user can browse through animations and explanations of the theoretical structures of the work including form, sound, harmony, melody, and rhythm/meter.

Bach's World: The composer's life and work are presented and the *Goldberg Variations* are put into an historical context.

The purpose of this software is to provide a resource that will enable non-linear inquiry into the structure, meaning, and sound of Bach's *Goldberg Variations.* The software presents large amounts of information in a compact form that is accessible any of a number of ways, but always connected with the sounds of the artwork. The interface invites exploration and experimentation. The program is appropriate for presentation in the classroom, or for private study outside of the classroom. Special emphasis is placed upon the incorporation of complete music notation so that the program is usable in college theory, history, and literature courses.

ERIC

The education hierarchy of computer-aided instruction and learning modules

AJS SUMMERLEE & JTG HOLMES
Associate Dean's Office
Ontario Veterinary College, University of Guelph, Guelph, N1G 2W1, Ontario, Canada.

The object of the demonstration/poster session will be to show that there is an hierarchy in the use of computer-aided teaching and learning modules. Computer-aided instruction (CAI) and computer-aided learning (CAL) modules can achieve different educational objectives and should be chosen for the particular needs of the teaching discipline.

In CAI modules, the amount and sequence of information that is to be learned is the domain of the teacher through design of the module. Learning is predominantly passive and the users enthusiasm and learning is excited by multi-media presentations. There is a need to motivate the student to drive the learning and this is usually achieved by integrating an examination or quiz into the module. The learning is compartmentalized and recall is usually confined to the circumstances of the initial learning experiences; while there is an increased possibility of integrated presentation of material, that information is usually pre-structured for the user.

More recently, the concept of computer-aided learning (CAL) has evolved. Here the module contains a resource base which can still be used as an instructional module but there is the possibility for student-directed search of the material and user structuring of the information. This means that the learning is more active, the module is less structured, the learning can be tailored to the needs of the particular user and the information can be learned integrated with the basic data. Effectively, the user has responsibility and control of the structuring of the information and this freedom, coupled with the breadth of enquiry that is possible creates a more exciting and motivating environment. Active exploration for information, establishment of contextual hooks to remember the data, and the ability to develop an integrated picture is more likely to lead to long-term memory.

Based on the educational objectives required in different teaching and learning scenarios, we have developed a number of learning modules using Toolbook, an authoring program by Asymetrix that runs under a Microsoft Windows Computing environment. Toolbook allows for the creation of a number of screens that combine text, graphics, animations and sound which are arranged as pages in a book. Hypertext links join the pages in any order desired. The modules make use of high resolution graphics obtained from scanned images of three-dimensional objects or photographs to illustrate particular points in the text. Different modules will be demonstrated to illustrate the different educational objectives that can be achieved with either CAI or CAL presentational formats.

The demonstration will provide hands-on experience of the flexibility of the software and to illustrate in practical terms the speed of the processing possibilities and the limitations and advantages of CAI and CAL in educational terms.

Developing a Cadre of University Faculty Multi-Media Experts

NANCY I. TODD
College of Education and Human Development, Mail Stop 90
Eastern Washington University, Cheney, WA 99004

Eastern Washington University was awarded a Title III grant from the U. S. Department of Education to develop a cadre of faculty experts in authoring and integrating multi-media into the university curriculum. Over a period of five summers, twenty-four faculty evenly selected from the university's six colleges will have participated, and, in turn, helped colleagues in their departments to develop multi-media instructional modules. The grant provides a summer salary stipend of $5000 for each faculty for product development and partial release time (equal to one course) for one quarter during the following academic year to field test the developed product.

There were two parts of the faculty selection process. First, applicants completed a self-assessment rating scale (1-6, unfamiliar to knowledgeable) for knowledge of Apple Macintosh, MS-DOS, authoring systems, Hypercard, videodiscs, CD ROM, multi-media availability in one's discipline, and understanding of effort required to develop a multi-media project. The second part called for a narrative limited to five double-spaced pages to support self-perceived strengths identified in five areas: (1) a skills statement that describes the extent of microcomputer skills as assessed in the first part; (2) an experience statement which explains previous and current involvement with multi-media; (3) a knowledge statement which details knowledge of multi-media available to one's academic discipline; (4) a project statement defining an undergraduate multi-media project to develop during the funding period; and (5) an understanding statement describing knowledge of the time and effort required to develop the multi-media project.

Initially, $30,000 was spent for equipment purchases, and another $5000 for software. Computer workstations included two NeXT workstations, one Macintosh IIsi, two Macintosh IIci, and two Zenith MS-DOS machines. In addition, two videodisc players, two CD-ROM drives, one cam-corder, two scanners, two projection panels with overhead projectors were purchased. Each year, two electronic demonstration stations on wheels will be acquired that include a Macintosh color computer, CD-ROM drive, videodisk player, speakers, projection pad. These will be distributed to each of the six colleges.

Faculty participated in summer workshops for eight weeks consisting of daily four hour sessions. Instruction in instructional design principles and authoring systems, such as *Hypercard*, *Macromind Director*, and *Authorware*, was provided. The university provided release time for a Computer Science faculty member to conduct the training workshops. Computer technicians also assisted in the workshop. Instructional design principles were followed that emphasized problem definition, task analysis, development of objectives and selection of appropriate instructional strategies and materials. Implication of research findings were applied to the design of instruction, as well as design of materials, taking into account such factors as learning hierarchies, instructional sequencing, graphic design, and screen display.

Participants were required to demonstrate and discuss their products at a university-wide symposium. These reports include evaluations of the effectiveness of both the training and the efficacy of integrating the multi-media project into the curriculum. First year faculty reported that the development project helped them to focus on ways to improve teaching. They found that they were re-considering the best ways to graphically represent information and to improve sequencing and verbal clarity. They also have increased confidence in their technical skills to now help colleagues develop multi-media applications. So far, projects have been developed for chemistry, composition, counseling, physics, language disorders, music, biology, and applied psychology.

Teaching Chinese as a Second Language: A Demonstration of *HyperChinese*

CHUN-HUNG TZENG
Department of Computer Science
Ball State University, Muncie, IN 47306, USA
and
LEON HSU
University Computing Services
Ball State University, Muncie, IN 47306, USA

Teaching Chinese as a Second Language (TCSL) in American schools is becoming more and more in demand because of the increasing interactions between China (PROC and ROC) and the U.S.A. in business, industrial, and cultural activities. It has been announced that Chinese will be included in SAT as one of the foreign languages. Teaching Chinese to American students requires human resources and learning resources. This hypermedia project is created by the presenters to meet the needs of learning resources for teaching and learning the Chinese language.

HyperChinese is a set of HyperCard stacks consisting of three components: *HyperFaYin*, *HyperXiZi* and *ChineseRadical*.

HyperFaYin is mainly for Chinese pronunciation of the Common Speech. It has three sections: 1) Text: The text database includes explanations on Chinese initials and finals with English translations of Chinese phrases and sentences. 2) Graphics: This section shows Chinese characters in graphic form with phonetic diagrams of animated movements of lips and tongues for all Chinese initials and finals, and four arrows for different pitch tones. 3) Sound Resources: This section has individual sound for each sentence for learning Chinese pronunciation. Users can play the whole conversation at once or each sentence individually.

HyperXiZi is a set of HyperCard stacks of Chinese characters mainly for practice Chinese writing. It also has three sections: 1) Text: It includes an English translation and the position for each stroke. 2) Graphics: It shows the Chinese character and its strokes in animated movements with adjustable speed. Users can practice writing on each outlined character with a paint brush. 3) Sound Resources: It gives a sound for each Chinese phrase formed by two characters.

ChineseRadical contains information of all Chinese radicals, including stroke orders, meanings, and pronunciations. The structure of the stack is organized very similar to that of *HyperXiZi with* the stroke order being the main part of the database.

The special features of *HyperChinese* may be summarized as follows: 1) Animation: The three types of information (text, graphics, and sound) can be retrieved jointly so that self-running "movies" are created. Each Chinese sound can be played with animated movements of lips and tongues, or with an animated pitch arrow. Each conversation can be played with the corresponding movements of its text. 2) Interactivity: The student can practice writing the Chinese character with a paint brush and compare it with the animated character stored in the database. 3) Developer's Tools: To facilitate the development, a set of tools was created. With these tools, any instructor of Chinese with minimal knowledge of HyperCard structure can easily load new lessons into the software. The new courseware can be developed semi-automatically. Therefore, the instructor can devote attention to the course contents rather than to the techniques of programming.

Hypermedia Environments in Medical Education

MARIO VEITL and PAOLO PETTA
Austrian Research Institute for Artificial Intelligence
Schottengasse 3, A-1010 Vienna, Austria

Medical education draws on a wide range of information sources such as pictures, sound, simulation, animation and video, and prostires in working with complex concepts. We demonstrate how hypermedia systems can provide valuable support by showing examples from courses developed in collaboration with the faculty of pathology of the University of Vienna. During the development of these courses we pinpointed primary causes for the problems encountered. From there we derived a generic guideline for the design of learning environments. This guideline refers to the overall structure of the curriculum, methods of presentation, and novel didactic approaches which are currently in a stage of rapid and continuous development.

First of all the coverage of the course has to be defined and analysed. From this follow both the contents of the course as well as the needed means of presentation. The course material is subsequently structured into lessons, for each of which clear goals must be defined. Each lesson consists of an introduction, a body, examples, and test items.

Students learn concepts more easily if they know precisely what they are to learn. Thus objectives must be defined clearly and accurately in the introduction. The body of a lesson prepares the users for the subsequent examples. In some cases users should be allowed to access the presentation first, in order to get a "feeling" for the domain and to be able to compare their own concepts and theories to the explanations given by the system. To support progress according to the paradigm of learning by exploration, the presentation includes examples and counter examples of a concept, annotated with explanatory comments. Finally, tests offer the opportunity to assess the learning progress. Feedback during a test should serve two main purposes: motivation to proceed and explanations of correct and incorrect answers.

As hinted at above, it is only after delimiting the contents of a lesson that the learning strategy should be decided upon. The options range from a tutorial-like guided approach to hypertext-like free browsing. In any case, appropriate orientation aids must be provided. In our examples we show the inclusion of shallow guides (maps representing the lesson structure) and the option of obtaining information about the current position within the course. Another important issue is the provision of a customizable environment. While support for private annotations is desirable, in our example we show how a comparatively simple notepad can already be used as a valuable tool.

The demonstrated courses were developed using the authoring tool Authorware Professional on Apple Macintosh computers (IIx and Quadra 900) equipped with digital video overlay cards (Mass Micro ColorSpace IIi/FX and VideoLogic DVA4000) using multistandard video disc recorders and players (Sony LDP3600D).

ERIC

SINERA on Disc: An Educational Hypermedia CD-ROM

JORDI VIVANCOS
Programa d'Informàtica Educativa (PIE)
Departament d'Ensenyament, Generalitat de Catalunya
Via Laietana, 64, 1er. 08003 Barcelona (Spain)
E-mail: VIVANCOS@DECNET.UPC.ES

SINERA on Disc is the first educational multimedia CD-ROM produced in Catalunya (Spain). Developed by the Programa d'Informàtica Educativa (current governmental plan of educational computing in primary and secondary education levels), it provides access to a wide range of educational resources in an integrated environment. The development process started in mid 1991 and the final CD-ROM was pressed on March 1993.

CD-ROM contents

-SINERA: bibliographic database of curriculum materials for primary and secondary education. Available on-line since 1987, SINERA contains at present more than 20.000 records;
-General thesaurus to assist the SINERA indexing and information retrieval processes;
-Multilingual dictionary (Aranes-Catalan-Spanish-French);
-Thesaurus of educational mathematics terms;
-90 educational software packages, including full-text user documentation;
-200 MIDI files and musical scores to play with MUSIC (educational music editor);
-A collection of digital maps for local geography studies.

All these materials are accessible in a dual way: from a hypermedia link within their own bibliographic record on SINERA database and by means of structured menus from the contents screen.

Interface functionalities

SINERA on Disc has been implemented using Multimedia Viewer 1.0 of Microsoft. This software runs on 386 and above computers under Windows 3.1 environment. The choice has been adopted after evaluating more than 20 information retrieval packages. Viewer was selected, basically, because of the possibility to manage in an integrated way an heterogeneous set of resources, its robustness and no programming requirements (although some good knowledge of Windows is needed), and the user friendly interface identical to the Windows on-line helps.

Windows and Viewer provide the interface and framework to browse, search, navigate, display and extract information from the application. Viewer is a software product oriented to produce electronic documents, such as on-line help (Windows 3.1 help has the same features as Viewer files), electronic reference books, on-line manuals and multimedia documents.

Since Viewer is not a "professional" information retrieval package for managing large databases, it offers limited search features: (numerical and field oriented search) and not customized output. For our target audience of primary and secondary education teachers and students these limitations are not critical.

ERIC

Design issues

The major challenge of this project was to convert a bibliographic database into a hypermedia database, giving access from the bibliographic reference to full-text documents, graphics, sounds and external educational software.

After evaluating several prototypes of SINERA on Disc, we adopted a dual format display for the database records: tagged format and graphical format. This led to the duplication of the database records, but CD-ROM offers a huge capacity!. The dual visualization format allows us to keep the former on-line record display as a "complete tagged record". For the second display format we have defined a new graphical record using the Windows large choice of shapes, colours and typeface sizes. The graphical record presents less information than the complete tagged record, but it offers increased readability, and its design intends to be a library card metaphor.

A choice of educational materials owned by the Department of Education of the Generalitat de Catalunya have been included on SINERA on disc. This decision enriches the bibliographic database with complete resources (full-text documents, software, graphics, MIDI sounds and "Video for Windows" videoclips) linked to the bibliographic record of the resource by a set of specific purpose icons. The range of Hypertext-hypermedia links between references and applications are: record-record (cross references), record-graphics (illustrations related to the record), record-sound (associated music), record-external software (run DOS and Windows programs from their bibliographic reference), record-full text document (Write and ASCII documents are displayed on screen using a specific editor), and record-video (a short sequence of the referenced video).

Final considerations

The volume of information contained on this first issue of SINERA on Disc is around 200 Megabytes. The distribution of this amount of information represents a cost-effective solution compared to other means of software and document dissemination. SINERA on Disc has been distributed free of charge during spring 1993 to more than 1000 primary and secondary schools in Catalonia.

SINERA on Disc will be updated once a year. We hope that this application will produce a catalytic effect on the learning and teaching opportunities and also on the local CD-ROM and multimedia industry.

References:

CD-CINC (a SIGCAT committee) (1991) CD-ROM Consistent Interface Guidelines, Final Report (Internet File)

Crown, J. (1992) Effective Computer User Documentation. Van Nostrand Reinhold

Horton, W. (1990) Designing & Writing online documentation. Wiley

ERIC

HyperLearning: Students Creating with Hypermedia

Louise Wilson, University of Minnesota
1276 Nursery Hill Lane, Arden Hills, MN 55112

Hypertext and multimedia products are wonderfully exciting instructional tools which make learning come alive. Laser disk and CD-ROM applications are becoming more and more commonplace in classrooms and learning centers. However, as wonderful as the completed products may be, even more powerful is the process of developing them. Developers of hypermedia products indicate that they gain tremendous knowledge in the domains of information in which they work.

This presentation explores the power of student use of a hypertext productivity tool for creating presentations. The potential for language development, critical thinking, creativity, organizational skills, and cooperative learning is considerable. Learning content in enough depth to be able to present it is an exercise of great value to students, and the process of presenting it via a computer adds additional incentive.

A hypermedia tool provides an exciting medium for students. Today's tools, such as HyperCard for the Macintosh, HyperStudio for the Apple and LinkWay or LinkWay Live! for the IBM allow nonporgrammers to create hypertext, multimedia presentations through the use of simple, pull-down menus. These tools allow students to become a partner with the computer in the creative and cognitive process of organizing and presenting material in new ways.

Hypertext tools provide opportunity for presentations that are not linear; hence students can be creative and in fact should be encouraged to see the nonlinear structure of content and organize it in conceptual ways as opposed to sequential ones. Groups of students can work together and combine multiple small projects into a larger whole. Students must organize material in a way that is sound, yet the final product can be as individual as the students who work on it. As students consider how the material can be organized, flow charts, concept maps and other visual devices help students concretely see how ideas fit together or flow from one to others. Even material which is familiar can be approached in a new way as students manipulate and organize it into a computer presentation. Visual and auditory multimedia pieces can also be added to reports.

In order to prepare content and organize it for a hypertext report, students must become very familiar with the content and its organization, become a "mini'expert". Experts have unique understandings of the domains of knowledge in which they possess expertise. In-depth learning of content creates a conceptual framework for understanding how the content is organized and for applying content knowledge to novel but related information. Navigating through a domain of knowledge with which one is highly familiar is a much different intellectual experience than the processing of new, unfamiliar information.

The underlying process of acquiring and organizing knowledge is the key to why this process is so valuable to student learning. Students of all ages and abilities can participate. Presentations can be verbal or nonverbal, simple or complex, short or long, graphic or textual. In all cases, giving students time to become expert in a content area, to organize material in conceptually interesting ways, and to create something exciting with the computer is a valuable use of educational resources and time.

Multimedia Experiential Recording: Making Reflective Artifacts for Learning & Collaboration

Kristina Hooper Woolsey & Charles Kerns
Advanced Technology Group
Apple Computer

The Video Bulletin Board is a networked, knowledge-sharing system which a fourth grade science class and scientists at the Exploratorium, a hands-on science museum in San Francisco, are using to communicate. Students record video clips showing weather phenomena which they encounter in their environment. Then the students combine the digitized video clips with their questions to create messages posted on the Video Bulletin Board. Other students and scientists respond with annotations and new video messages. Each message show the portraits of the originator and the annotators to a message. The students also recorded video clips to document and ask questions about their discoveries on a field trip to the museum.

The system has multiple displays (consisting of a Macintosh computer controlling a video camera, video deck, and video digitizing card) in the classroom and at the museum. Each display can access and update a common database.

This system allow students and teachers to take advantage of emerging media-rich technologies in sharing their personal experiences, and then using these for reflections on conceptual issues.

ERIC

Author Index

ERIC

❖

Artificial Intelligence in Education:

Proceedings of the World Conference on AI in Education: 1993

EDITED BY STELLAN OHLSSON, ET AL.

August 1993, 475 pages (approx.); ISBN 1-880094-08-8

This book serves as a major source document indicating the current state of artificial intelligence in education and related fields.

This volume also serves as a record of AI-ED 93—World Conference on Artificial Intelligence in Education, held in Edinburgh, Scotland, in August 1993. The conference is one in this series of biennial international conferences designed to report the best research in the field of AI and Education and to provide opportunities for the cross-fertilisation of information and ideas on research and applications in this field.

Articles cover the following major topics:

- Intelligent tutoring systems
- Learning environment and microworlds
- Visual and graphical interfaces
- Human factor and interface design
- Intelligent multimedia systems
- Authoring system and tutoring shells
- Collaboration tools
- Training job skills
- Principle for instructional design
- Natural language interfaces
- Knowledge representation
- Knowledge and skill acquisition
- Conceptual change
- Metacognition
- Social and cultural aspect of learnin',
- Cognitive development and errors
- Student modeling
- Teaching higher-order thinking skill
- Theories of teaching
- Motivation
- Reading and writing
- Computer-assisted language learning
- Evaluation of computer system
- Assessment of learning outcomes

Ordering Information:

Non-member Price: $45
AI-ED Society & AACE Member Price: $35
Add $3.50 postage/handling (USA) and $8.50 elsewhere
Check (US funds), PO, VISA, and MasterCard accepted.

Available from:

AACE
P.O. Box 2966
Charlottesville, VA 22902 USA
804-974-3987; Fax: 804-978-7449; E-mail: AACE@virginia.edu

BEST COPY AVAILABLE

Technology and Teacher Education Annual, 1993

Doris Carey, Regan Carey, Dee Anna Willis, and Jerry Willis (Editors)
March 1993, 756 pages; ISBN 1-880094-05-3

This book contains over 150 papers presented at STATE 93—Fourth Annual Technology and Teacher Education Conference, the meeting of the Society for Technology and Teacher Education (STATE), a division of the Association for the Advancement of Computing in Education (AACE).

This Annual, as did the conference, explores the 1993 conference theme—*Technology Across the Curriculum*—and serves as a major source document indicating the current state of teacher education and information technology.

Described within this book are outstanding examples of successful technology integration in teacher education. Highlighted are applications as diverse as CAI, hypermedia, interactive videodisks, simulations, telecommunications, and much more. The applications also represent the range of instructional and learning theories and many levels of thought from "how to" articles to "context" papers, issue papers, and theory papers.

Collectively the papers demonstrate that teacher education, as a technology-using field, is thriving. There are validated models, methods, and instructional packages for everything from a classroom management simulation to a state-wide telecommunications system. This book chronicles the pioneering efforts of several hundred technology leaders in teacher education. Also included are seven papers receiving 1993 STATE awards for outstanding achievement.

Major book sections are:

- Diversity
- Multimedia
- Technology Diffusion
- Telecommunications
- Preservice Teacher Education
- Integrating Technology into Methods Classes
- Simulations
- Instructional Technology
- Research
- Inservice & Graduate Education
- Instructional Design
- Concepts & Procedures
- Mathematics & Science

Ordering Information: Non-member Price: $45
STATE & AACE Member Price: $35
Add $3.50 postage/handling (USA) and $8.50 elsewhere
Check (US funds), PO, VISA, and MasterCard accepted.

Available from:
AACE
P.O. Box 2966
Charlottesville, VA 22902 USA
804-974-3987; Fax: 804-978-7449; E-mail: AACE@virginia.edu

BEST COPY AVAILABLE

BEST COPY AVAILABLE
ERIC

AACE

ASSOCIATION FOR TH: ADVANCEMENT OF COMPUTING IN EDUCATION

ISBN 1-880094-06-1

BEST COPY AVAILABLE