DOCUMENT RESUME

ED 396 696 IR 017 872

AUTHOR Williams, Marian G.; Traynor, Carol
TITLE Participatory Design of Educational Software.
PUB DATE 94
NOTE 7p.; In: Recreating the Revolution. Proceedings of the Annual National Educational Computing Conference (15th, Boston, Massachusetts, June 13-15, 1994); see IR 017 841.
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Case Studies; *Computer Software Development; Computer Uses in Education; *Educational Technology; Engineers; Guidelines; High Schools; Instructional Materials; Material Development; Research and Development; Secondary Education; *Teacher Participation; *Users (Information)
IDENTIFIERS Massachusetts (Boston); *Participatory Design

ABSTRACT

The emerging field of participatory design is devoted to involving end users of a new technology in the design of that technology. This paper presents a case study of the participatory design of educational software and discusses guidelines derived from federally-funded formal research on participatory design projects. The case study involved the introduction of new computer technology (a newspaper editorial system) into the curriculum of a Boston public high school English department, which previously had no computer technology available in the classroom. In order to adapt the editor-writer model for the classroom, the editorial system needed to be customized. Customizations fell into three categories: information flow, security, and usability. While the design team included both teachers and engineers, the design effort was lead by a former English teacher turned computer scientist. This "translator" enabled the teachers to participate in the customization process by mediating between their language and workplace conventions and those of the software engineers. The formal retrospective study examined this project as well as another project that involved introducing new technology into the classroom. Guidelines for future participatory design projects were derived from interviews of the project participants, which included teachers, administrators, university researchers, and corporations. (Contains 15 references.) (AEF)

Paper (W4-201B)

Participatory Design of Educational Software

Marian G. Williams
Center for Productivity Enhancement
University of Massachusetts Lowell
Lowell, MA 01854
(508) 934-2630
williams.chi@xerox.com

Carol Traynor
Computer Science Department
University of Massachusetts Lowell
Lowell, MA 01854
(508) 934-3630
ctraynor@cs.uml.edu

Key words: participatory design, software design, computer-based educational technology

Abstract

This paper looks at the problem of giving public school teachers a real voice in the design of educational software. We believe that the solution will be found in the emerging field of participatory design, which seeks techniques for involving workers in the design of new technology that will change the way they do their jobs. We present a case study and describe our federally-funded research project on participatory software design with teachers.

Introduction

At a recent panel discussion about enabling users to participate in software design (Williams, Begg, Kuhn, Richardson, & Suchman, 1993) a member of the audience responded to a description of the difficulty of enabling computer-illiterate schoolteachers to participate in the design of instructional software by demanding, "What are the teachers doing about it?! What initiative are they taking?!" We believe that teachers carry a double burden — not only the all-to-common disregard of computer professionals for the users of their software, but also the all-to-common failure of the general public to acknowledge that teachers have many of the same workplace issues as other workers.

In this paper, we look at public school teachers as workers and at the public school as their workplace. We consider teachers to be the arbiters of curriculum. Thus, we view as crucial the task of learning how to give teachers, especially those who are not computer-literate, a real voice in the design or customization of the software they will use on the job — that is, in the classroom. We believe the answer to giving teachers that real voice will be found in the emerging field of participatory design.

Participatory design is devoted to involving the end users of a new technology (such as new software) in the design of that technology — especially in cases where the technology will change the way the workers do their jobs (Schuler & Namioka, 1993). The underlying philosophy of participatory design is that it is at best arrogant and at worst impossible for outsiders to predict what impact a new technology will have on someone's work and workplace. Participatory software design differs from traditional approaches to collaborative software design in that it seeks to give workers a direct voice in design, rather than to have a computer professional speak on their behalf. In particular, it seeks to secure workers' participation beginning with the earliest stages of design, well before the formative evaluation of software or prototypes developed by engineers. Researchers are studying how to involve workers as diverse as nurses (Bjerknes & Bratteteig, 1987), architects (Peng, 1992), and software engineers (Muller, 1991) in the design of new technology. Little participatory design work, other than our own, has focused on teachers (Williams, 1993; Williams & Begg, 1992; Williams & Begg, 1993a; Williams & Begg, 1993b).

Clement and Van den Besselaar took a retrospective look at participatory design projects (Clement & van den Besselaar, 1993). Their survey included projects that took place in a variety of countries and a variety of workplace settings. They observe that the five most important ingredients in a participatory design project are these: (1) workers must have access to information that is relevant to the project; (2) it must be possible for the workers to take an independent position on the problems; (3) workers must participate in the process of decision making; (4) participatory development methods must be available for use by the participants; (5) there must be organizational and technical flexibility, not an expectation that the participants will adhere strictly to a pre-determined organizational model or technology. In our experience, the third and fourth of these ingredients are the most crucial in the workplace of the public school teacher.

Workers must participate in the process of decision making. In many parts of the country, including Massachusetts, public school faculties have shrunk because of budget cuts and changing enrollments. The youngest teachers, the ones most apt to have had computer training, have been laid off. The remaining teachers, in many cases, have little or no familiarity with computers. For example, in the case study described in the next section, only two of the teachers we worked with had substantial experience using a computer, and their experience was limited to word-processing. It is as difficult in the teachers' workplace as in any other workplace to achieve true user participation in software design or customization.

Participatory development methods must be available for use by the participants. It is our opinion that the key issue here is translation between users and software developers. The translation involves not only the different terminology used by teachers and software developers, but also the understanding of each other's work and workplace. We have successfully used a former-teacher-turned-computer-scientist as the translator in our projects. The translator understands not only the language used by teachers and by software developers, but also the detailed nature of their work and conventions of their workplaces. The translator can employ participatory design techniques to make sure that there is a meeting of the minds between teachers and software developers.

For general information on participatory design, see (Muller & Kuhn, June 1993; Muller, Kuhn, & Meskill, 1992; Namioka & Schuler, 1990; Schuler & Namioka, 1993). For more information about translation in participatory design, see (Williams & Begg, 1992; Williams & Begg, 1993a; Williams & Begg, 1993b).

The next section presents a case study of the participatory design of educational software. It is followed by a discussion of the conclusions drawn from the case study and a description of our federally-funded research project for the formal study of the participatory design of educational software.

BEST COPY AVAILABLE

Case study

This case study concerns the introduction of new computer technology (a newspaper editorial system) into the curriculum of a public-school English department, which previously had no computer technology available in the classroom. The customization of the new software was an exercise in participatory design.

An editorial system is a hardware and software network used by reporters and editors at a newspaper. It is designed especially for creating, sharing, and editing newspaper articles. An editorial system is highly customized for writing and editing, and for collaboration between writers and editors. The newsroom staff create the content of articles, but are not responsible for the appearance of the final page. Thus, an editorial system provides basic text-formatting capabilities, such as bold and italic, and does not include a page layout facility.

The hardware and software for a 22-seat editorial system were donated by Atex Publishing Systems (Billerica, MA, USA) for this project. The editorial system has been installed in the public high school of a middle-class suburb of Boston. The school has 10 English teachers and an enrollment of 900 students in grades nine through twelve. The editorial system has been customized for teaching and learning, and is currently in its second semester of use at the high school. To our knowledge, this project is the first to customize a commercial newspaper editorial system for instructional use in the high school classroom.

The editorial system's hardware consists of 22 terminals (a mixture of Atex terminals and IBM PC's); Atex proprietary keyboards, which are highly customized for writers and have dedicated or programmable keys for many word-processing functions; a customized Digital Equipment Corporation minicomputer; and a laser printer. The software is Atex's basic editorial system, used worldwide by newspaper writers and editors. The software offers traditional word-processing capabilities, nearly all of which are available via special keys on the keyboard. It also offers features for cooperative work by writers and editors. These features include electronic mail, file sharing, and red-lining.

Each sophomore and junior English class uses the editorial system lab, instead of a traditional classroom, for one out of four terms. Thus, the lab is used by a total of 450 students per year. All of the English department's 10 teachers chose to be trained to use the editorial system.

The premise of the project is that the interaction between teachers and students in the writing classroom can mimic in may ways the interaction between editors and writers at a newspaper. However, a classroom is a different workplace from a newsroom. In particular, the flow of information in the classroom is different from the flow of information in a newsroom. In order to adapt the editor-writer model for the classroom, the editorial system needed to be customized. The types of customizations that were made are discussed below.

Participatory design of customizations to the editorial system. The teachers were experts in teaching writing, not in using computers. Understandably, none of the teachers knew enough about hardware or software to direct the customization of the editorial system or even to carry on useful dialogues with the software engineers. They had neither the skills to evaluate the existing capabilities of the editorial system nor the skills to tell the software engineers what customizations they needed. They were unfamiliar with basic computer-ese, such as "file," "directory," "username," and "electronic mail." They did not have the skills to look at differences in workflow in a paper-based vs. a computer-based classroom. Moreover, they were given no release time to learn about computers or to work on the customizations.

The engineers, on the other hand, were experienced in tailoring editorial systems for specific newspaper sites. They were knowledgeable about the editorial system and about workflow in a newsroom. They were used to newspaper jargon. (In fact, a lot of that jargon had been incorporated into the editorial system. Where a teacher would "throw away a paper" and a computer person would "delete a file," a newspaper person would "spike a story," a metaphor for the traditional metal spike on which papers to be discarded could be impaled. In the editorial system, a file is deleted by sending it to the "spike queue.") The engineers' language was a mixture of computer talk and newspaper talk, while the teachers spoke the language of writing instruction.

Having spent time in the classroom as students, the engineers had general assumptions about what teachers do, but were not aware of specific activities and conventions of the workplace. (A trivial example of their unfamiliarity with life in the high school classroom was their suggestion to use "stud" as an abbreviation for "student" in naming some computer accounts.) Their job was to perform customizations, but they had neither sufficient time nor sufficient knowledge of the teachers' workplace to lead the design of those customizations.

While the design team included both the teachers and the engineers, the design effort was lead by a university researcher who was a former English teacher turned computer scientist. This "translator" enabled the high school teachers to participate in the customization process by translating between their language and workplace conventions and those of the software engineers. The translation process had these steps: work with the teachers to develop a description of writingrelated activities currently used in their paper-based classrooms, including workflow and paperflow; determine how these activities could be carried out and extended, based on the existing and potential capabilities of the editorial system; figure out the implications of the computer-based versions of these activities on information flow, security, and usability; teach the teachers, in their own language and in terms of classroom activities, about features that would extend their writing-instruction activities, in order to enable them to decide if they wanted the extensions; translate the descriptions of the activities from the language and workplace conventions of the teachers to the language and conventions of the engineers: and work with the engineers to specify the customizations to the editorial system.

The customizations fall into three categories: information flow, security, and usability. They address issues such as how to support the teaching of process writing, how to discourage plagiarism, and how to help students have easy access to the materials they need to have at hand. An example of each follows.

In a typical newsroom, there is only one current version of a "story." Either the editor has it or the writer has it. When one of them sends the story to the other, the story disappears entirely from the sender's workspace and now appears only in the recipient's workspace. Earlier versions of the story are stored in an archive and are rarely retrieved. By contrast, in an English classroom where process writing is taught, students are expected to maintain a library of the various drafts of an essay. When a student submits a draft to the teacher, the student should retain all earlier versions, as well as a copy of the current version, in his or her workspace. The versioning capabilities of the editorial system were customized to reflect these differences between the newsroom and the classroom. When a student submits a paper to the teacher electronically, a new version number is automatically assigned, and all drafts are archived in the student's own workspace.

The Atex engineers tell us that plagiarism is not a major issue within the newsroom. Consequently, the editorial system permits writers to have access to stories in each other's workspaces. In addition, old versions of stories are stored in a public archive and can be retrieved by any writer or editor. Plagiarism is, unfortunately, a concern in the writing classroom. The editorial system was customized to remove the temptation for plagiarism. A student is given access only to his or her own files and to certain public files made available by the system manager or the teacher. Moreover, the public archive of previous versions of essays was eliminated. All drafts of a student's essays are kept in the student's own workspace; students do not have access to each other's workspaces. (It is a basic feature of the editorial system that a document that is mailed from one user to another carries the creator's name with it in a system-maintained header.)

In the paper-based classroom, the teacher hands out a variety of department guidelines, such as the guidelines for acceptable manuscript form, the late work policy, and the glossary of correction symbols. Students are required to have these documents at hand during class. The on-line help facility of the editorial system was customized to include these guidelines. From the main help menu, students can go directly to any of these official department documents, thus obviating the need for each student to carry the documents to class.

We have not yet conducted a formal study of the effectiveness of the editorial system for teaching and learning writing. For now, we can report only anecdotal evidence of the success of this project. The teachers report the usual motivational value of successfully integrating computers into the curriculum, in particular that students enter the lab and get right to work rather than chatting with their classmates, that non-attending students start attending class, and that students who appear passive in the traditional classroom are actively engaged by the computers.

In addition, the teachers report that students are giving greater attention to the writing task. The teachers are besieged with questions about style, organization, and usage. Moreover, the teachers report that the comments that they embed in a student's electronic essay are taken more seriously than comments written in the margins of an essay submitted on paper. They say that students are more apt to perform the revisions that the teacher requests. A couple of the teachers had previously watched students use Apple computers, and had observed how the students were consumed by the joys of using various fonts, type sizes, and special effects for making their text look dramatic. With the editorial system, they report that the students are focused on making the writing good, not on making it gorgeous.

As with many other efforts to computerize the writing classroom, including the majority of those described in (Wresch, 1991), the editorial system is being used to carry out traditional classroom activities that have been moved from chalkboard or paper to the computer. The teachers are gradually learning to exploit the editorial system's capabilities for collaborative work, through the sharing of electronic documents and messages, the use of redlining, etc.

Discussion

Customization of software for the classroom can be labor-intensive. The process of customizing the editorial system for use in the secondary English classroom required the resources of our university, of Atex Publishing Systems, and of the public school system. Those resources continue to be brought to bear for further curriculum development, hardware and software maintenance, and system management.

A member of the university research team served as the translator who lead the design of customizations. Atex provided engineers to install the editorial system and to perform the customizations, as well as a trainer to teach the teachers how to use the editorial system. The public school system bore the costs of site preparation, including the installation of a security system, and provided a summer workshop for revising the writing curriculum. The teachers have devoted countless hours to learning to use and to teach with the editorial system.

It is our opinion, based on comparison with another of our projects (Williams, Theriault, Stowe, & Canning, 1992), for which general-purpose hardware and software were used, that customization of special-purpose hardware and software poses different challenges. Because special-purpose commercial hardware and software, such as the editorial system, embody a way of doing work, they must be carefully tailored to the teachers' workplace and to the teachers' way of doing work.

The design team needs to include someone with the skills and resources to translate between the teachers and the engineers. The translator was vital to the customization of the editorial system, since the teachers and engineers did not have knowledge of the each other's work and workplace, and did not have sufficient resources to develop such knowledge. The translator was accepted by both teachers and engineers as somebody who spoke their language and who knew how their work responsibilities were being affected by the project. We believe that studying the contribution made by the translator will reveal additional techniques for empowering teachers to participate in software design.

Formal study of participatory design with teachers

With support from the U.S. Department of Education, we have embarked on a formal retrospective study of two of our projects that introduced new technologies into the classroom (the editorial-system-in-the-classroom project described above and the Revitalizing High School Computer Science project described in (Williams, Theriault, Stowe, & Canning, 1992). The goal of the retrospective study is to formulate a set of guidelines for the participants in future projects. These participants include not only teachers, but administrators, corporations, and university researchers, as well.

The retrospective study involves formal interviews with the teachers who participated in the projects (both those who championed the projects and those who did not); administrators at the school and school-system levels; corporate personnel who worked on the projects (including upper management, technical staff, trainers, and maintenance staff); and university personnel.

The guidelines for teachers focus on contributions that they can make to the design process without knowing a lot about computers. In our experience, teachers who are unfamiliar with computers feel inadequate for the design task, until they are guided to describe their classroom activities in their own language and in terms of their own workplace conventions. The guidelines should help teachers develop realistic expectations about their time commitments for the project. The guidelines treat teachers as the authorities on curriculum.

The guidelines for administrators focus on the ways administrators can provide support for teachers who are learning new technologies, participating in their customization, and teaching an innovative curriculum for the first time. In addition, the guidelines focus on the financial obligations that a school district can expect to incur, even with a "free" donation, and on the school district's interactions with university researchers and corporations.

The guidelines for university researchers focus on techniques for involving teachers in participatory design and on the development and management of a project of this type. They focus on the parts of the project that neither teacher nor corporate representatives may be able to do: translate between the teachers and the engineers. They also stress the importance of finding committed teachers who can champion innovative programs within their schools.

The guidelines for industry representatives focus on the obligation that must accompany a corporate donation of hardware and software. Even with the involvement of a university team, a corporation must expect to include installation, training, customization, and maintenance in their donation. Moreover, the corporation must schedule their personnel to spend time in participatory design sessions with the teachers and researchers. The guidelines explicitly address a problem that we have observed repeatedly: software engineers assume they know what goes on in a public school classroom, but they do not; as a result, they may make inappropriate design decisions.

The guidelines will be used in a new project, in which an innovative mathematics curriculum will be introduced into an inner-city school. Data will be collected both by observation during the design process and by surveying and interviewing participants after the design process. Moreover, the design documents (both those we provide and others devised during the design process) will be studied. The guidelines will be revised to reflect the lessons learned from the mathematics project, and will then be distributed to interested schools, universities, and corporations.

Summary

Participatory design is an important new approach to the development and customization of instructional software. Certainly, some teachers are capable of developing their own software and some off-the-shelf educational software can be incorporated into a given school's curriculum without customization. In our experience, many innovative uses of computers in the classroom are exceptions to these rules. For the use of computer technology in the classroom to move successfully in new directions, teachers as well as computer professionals must have a real voice in the design of that technology and in its incorporation into curriculum.

Acknowledgments

We are grateful to Atex Publishing Systems, Inc. for their contributions to this project and their enthusiasm for making it happen. We would like to express our admiration for the teachers we have worked with for their courage in embracing the editorial system as a new teaching tool and for their determination to incorporate it effectively into their curriculum.

This research is funded in part by U.S. Department of Education grant number R117E30237, "Improving the Success Rate of Adapting Technology for the Classroom."

References

Bjerknes, G., & Bratteteig, T. (1987). Florence in Wonderland: System Development with Nurses. In G. Bjerknes, P. Ehn, & M. Kyng (Eds.), Computers and Democracy: A Scandinavian Challenge (pp. 279-295). Aldershot, England: Avebury.

Clement, A., & van den Besselaar, P. (1993). A Retrospective Look at PD Projects. Communications of the ACM, Special Issue on Participatory Design, 36(6), 29-37.

Muller, M. (April 27-May 2,1991). PICTIVE—An Exploration in Participatory Design. In Proceedings of Human Factors in Computing Systems (CHI '91), (pp. 225-231). New Orleans, April 27-May 2, 1991.

Muller, M., & Kuhn, S. (Eds.). (June 1993). Communications of the ACM, Special Issue on Participatory Design, 36(6).

Muller, M., Kuhn, S., & Meskill, J. (Eds.). (1992). PDC '92: Proceedings of the Participatory Design Conference. November 1993, Cambridge, MA.

Namioka, A., & Schuler, D. (Eds.). (1990). PDC '90: Conference on Participatory Design. Seattle, WA, March 1990.

Peng, C. (1992). Participatory Architectural Modeling: Common Images and Distributed Design Developments In PDC '92: Proceedings of the Participatory Design Conference, (pp. 171-180). Cambridge, MA, November 6-7, 1992.

Schuler, D., & Namioka, A. (1993). Participatory Design: Principles and Practices. Hillsdale, NJ: Erlbaum.

Williams, M. G. (1993). Adapting a Commercial Newspaper Editorial System for Teaching and Learning. In Proceedings of the Association for Media and Technology in Education Conference (AMTEC '93), (pp. 158-163). Windsor, Ontario, June 13-16, 1993.

Williams, M. G., & Begg, V. (1992). Translation in Participatory Design. In PDC '92: Proceedings of the Participatory Design Conference, (pp. 113-14). Cambridge, MA, November 6-7, 1992.

Williams, M. G., & Begg, V. (1993a). Translation between Software Developers and Users. Communications of the ACM, Special Issue on Participatory Design, 36(6), 102-103.

Williams, M. G., & Begg, V. (1993b). Translation in Participatory Design: Lessons from a Workshop. In Adjunct Proceedings of INTERCHI '93, Amsterdam, April 24-29, 1993.

Williams, M. G., Begg, V., Kuhn, S., Richardson, C., & Suchman, L. (1993). Participatory Design — The Natural History of Communication. Panel presentation at Greater Boston SIGCHI, July 15, 1993.

Williams, M. G., Theriault, C., Stowe, A., & Canning, J. T. (1992). Revitalizing High School Computer Science. In Proceedings of the National Educational Computing Conference (NECC '92), (pp. 181-186). Dallas, TX, June 1992.

Wresch, W. (Ed.). (1991). The English Classroom in the Computer Age: Thirty Lesson Plans. Urbana, IL: National Council of Teachers of English.

DOCUMENT RESUME

ED 396 697 IR 017 873

AUTHOR	Wright, Peter
TITLE	Let's Not Let the Number of Warthogs Be "X."
PUB DATE	94
NOTE	7p.; In: Recreating the Revolution. Proceedings of the Annual National Educational Computing Conference (15th, Boston, Massachusetts, June 13-15, 1994); see IR 017 841.
PUB TYPE	Reports - Descriptive (141) -- Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Cognitive Style; Computer Uses in Education; Educational Development; *Educational Technology; Elementary Secondary Education; Foreign Countries; Information Technology; *Mathematics Education; *Problem Solving; *Spreadsheets; *Word Problems (Mathematics)

ABSTRACT

The issue of how to integrate information technology in the teaching/learning environment remains strongly associated with the use of the computer as a tool. While technology based tools such as Logo have been advocated for problem solvers at the elementary level, spreadsheets have a great deal of potential for use at both the junior and senior high school levels. In this paper, alternatives to the solution of a variety of math problems, ranging from story problems to finding the root of equations, are explored. Although not all problems lend themselves to the use of spreadsheets, such an approach does add a viable alternative to existing problem solving strategies. The greater the number of strategies available to students, the more likely it is that differences in learning styles can be accommodated. Seven figures depict solutions to the sample problems. (Author)

Paper (W4-202A)

Let's not let The Number of Warthogs be 'X'

Peter Wright
Department of Adult Career and Technology Education
Faculty of Education
630 Education Building South
University of Alberta
Edmonton, Alberta
Canada T6G 2G5
(403) 492-5363
peter_wright@act.educ.ualberta.ca

Key words: problem solving, high school, spreadsheet, strategies, microcomputer

Abstract

The issue of how to integrate information technology into the teaching/learning environment remains strongly associated with the use of the computer as a tool. While technology based tools such as Logo have been advocated for problem solvers at the elementary level, spreadsheets have a great deal of potential for use at both the junior and senior high school levels. In this presentation, alternatives to the solution of a variety of math problems, ranging from story problems to finding the roots of equations, will be explored. Although not all problems lend themselves to the use of the spreadsheet, such an approach *does* add a viable alternative to existing problem solving strategies. The greater the number of strategies available to students, the more likely it is that differences in learning style can be accommodated.

Introduction

It is now approximately fifteen years since micromputers appeared in the schools; many interesting phases have been witnessed since the inception. At first, in the 'euphoric phase', attention focussed on the issues of what type of computers to buy, how many there should be, and equality of access. Sheingold (1991) acknowledged this phase quite recently in remarking that "computer-based technology has been brought into schools during the past decade largely because the technology was seen as being important in and of itself". Attention next turned to "but what do we do with them" and in the absence of quality educational software, a widespread computer programming epidemic broke out thereby marking the 'dawn of reality' phase. Fortunately, during this period, productivity and general purpose software emerged. This eventuality marked a very significant downturn in the popularity of computer programming and gave rise to the 'exploitation phase' during which the computer could be employed as a general purpose tool by teachers and students alike. Despite the passage of time and rapid advances in information technology, the 'electronic education phase' that visionaries had predicted is still not a pervasive reality. The burning issue of how to integrate information technology into the teaching/learning environment thus remains strongly associated with the use of the computer as a tool. This presentation will describe and demonstrate a number of ways in which the spreadsheet can be exploited in the mathematics classroom and through them, examine the problem solving strategies available to students. An assumption is made that students have been introduced to spreadsheet basics.

Of Warthogs and Cockatoos

The first example presented represents the classic story problem an example of which is as follows:

> The total number of legs in a group of 14 animals is 38. The group contains only cockatoos, which have 2 legs each, and warthogs, which have 4 legs each. How many warthogs are there?

The traditional approach to solving a story problem of this type is to begin by saying "let the number of warthogs be X" then proceed to establish and solve a set of simultaneous equations. Such a rigorous, analytical strategy can be very appealing to the mathematically inclined but less so to those who are not. The less mathematically inclined might choose to make an inspired estimate of the number of animals of each animal type, determine how many legs are implied and then adjust their estimate until they zero in on the answer—the trial and error method. Both strategies are perfectly valid. The spreadsheet offers a number of middle-ground alternatives which serve to widen the spectrum of problem solving strategies available to students. Three potential strategies that students might employ in solving the warthogs and cockatoos problem are described below in order of increasing level of sophistication.

Strategy 1

One of the simplest anticipated solutions might contain three columns as shown in Figure 1. Column 'A' contains an increasing sequence of natural numbers from one to fourteen (corresponding to the potential number of cockatoos in the group of animals); the students may either enter these numbers directly or generate them using a simple formula as shown. Column 'B' contains the corresponding number of warthogs (fourteen minus the number of cockatoos); these numbers may also be entered directly or generated by a formula. Column 'C' uses a formula to calculate the total number of legs for the fourteen animals (two times the number of cockatoos plus four times the number of warthogs). While this formula could be entered into each cell, students should be expected to be familiar with the simpler concepts of copying formulas between cells.

	A	B	C
1	#COCKATOOS	# WARTHOGS	TOTAL LEGS
2	----------	----------	----------
3	1	13	54
4	2	12	52
5	3	11	50
6	4	10	48
7	5	9	46
8	6	8	44
9	7	7	42
10	8	6	40
11	9	5	38
12	10	4	36
13	11	3	34
14	12	2	32
15	13	1	30
16	14	0	28
17			

=A15+1 =14-A16

=(A16*2)+(B16*4)

Figure 1.
Warthogs and cockatoos—strategy 1

The answer to the problem is obtained by scanning down column 'B' to locate the cell where the total number of legs is 38 (row 11 in this example); the answer to the problem is then derived from the corresponding cell in columns 'A' i.e. there are *5* warthogs.

Strategy 2

The solution shown in Figure 2 reflects an entirely different way of thinking about the same problem. As with solution 1, column 'A' is filled with the natural numbers from one to fourteen to represent the number of animals. In column 'B', each animal is given two legs because each animal type represented in the group has at least two legs (thereby accounting for the first 28 legs). This column can be filled manually or using a simple incrementing formula. The "leftover legs" are next allocated two at a time in column 'C' thereby "creating" the four legged animals. Cell 'C17' contains a formula which calculates and displays the running total of the legs allocated. When this total equals thirty eight, one simply counts the number of animals which have an extra pair of legs—this will be the animal number read from column 'A' corresponding to the last entry in column 'C'.

	A	B	C
1	ANIMAL #	INIT. LEG ALLOCN	EXTRA LEGS
2	=======	=============	========
3	1	2	2
4	2	2	2
5	3	2	2
6	4	2	2
7	5	2	2
8	6	2	
9	7	2	
10	8	2	
11	9	2	
12	10	2	
13	11	2	
14	12	2	
15	13	2	
16	14	2	
17		28	38

=Sum(B3:B16)   =Sum(B3:C16)

Figure 2.
Warthogs and cockatoos—strategy 2

Strategy 3

The solution shown in Figure 3 is an automated version of solution 2 and is presented here to illustrate the potential variety of student approaches which might be anticipated. Column 'A' and column 'B' are filled in one of the ways described previously. The first entry in column 'C' contains the number "2". Cells 'C4' to 'C16' contain a formula which automatically allocates an extra pair of legs to the animals until all 38 legs have been assigned. The answer is read in the same manner as for solution 2.

	A	B	C
1	ANIMAL #	INIT. LEG ALLOCN.	EXTRA LEGS
2	=======	=============	========
3	1	2	2
4	2	2	2
5	3	2	2
6	4	2	2
7	5	2	2
8	6	2	0
9	7	2	0
10	8	2	0
11	9	2	0
12	10	2	0
13	11	2	0
14	12	2	0
15	13	2	0
16	14	2	0
17		28	38

=If(B17+Sum(C3:C3)<38,2,0)

Figure 3.
Warthogs and cockatoos—strategy 3

This third solution is the most sophisticated of the three that have been presented. This solution entails the use of absolute cell references and the "logical if" function—it clearly requires a higher level of proficiency with the spreadsheet.

Finding the Roots of Equations

In mathematics, students are taught a variety of methods of "solving" (or finding the roots of) equations ranging from factoring to synthetic division to graphical analysis. Very often, however, equations do not have "nice roots" thereby lessening the convenience of algorithmic methods. In these instances, a simple spreadsheet (with or without plotting capability) can provide the learner with a viable tool for exploring the roots of equations by iteration. Figure 4 shows how this can be accomplished for a particular quadratic equation. As well, the approach described can provide the teacher with a very useful demonstration tool which allows for relatively quick and easy simulations.

ROOTS OF EQUATIONS (SS)

	A	B	C	D	E	F	G
1	Iteration Table			Start with X:		1	
2	X	Y		Increment by		0.1	
3	====	=======					
4	1	-2.0					
5	1.1	-1.7					
6	1.2	-1.4		Roots of $Y=X^2+X-4$			
7	1.3	-1.0					
8	1.4	-0.6					
9	1.5	-0.2					
10	1.6	0.2		contains A9*A9+A9-4			
11	1.7	0.6					
12	1.8	1.0					
13	1.9	1.5					
14	2	2.0					
15	2.1	2.5	X	1	1.1	1.2	1.3
16			Y	-2	-1.69	-1.4	-1.01

Figure 4.
Using a spreadsheet to estimate the roots of an equation

Using the graphical capability of the spreadsheet to plot the first iteration yields the graph shown in Figure 5. It is clear from the iteration table and reinforced by the graph that one root lies between "1" and "2" and the other lies between "-2" and "-3".

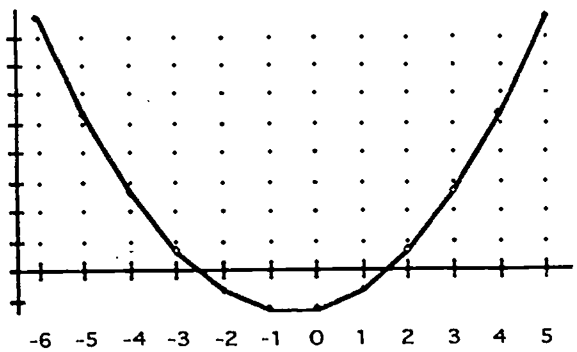

Figure 5.
Graph of $Y=X^2+X-4$, first iteration

From the graph of the first iteration, one of the roots is estimated to be x=1.5. If this root is required to further precision, a finer iteration can be carried out. The graph of such an iteration is shown in Figure 6—the start point and increment have been adjusted to zero in on the root.

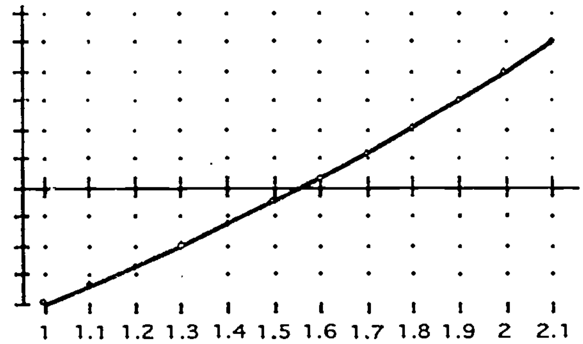

Figure 6.
Graph of $Y=X^2+X-4$, second iteration

From the second iteration, this root can now be estimated to two decimal places (x=1.56).

Towards Winning the State Lottery

This example, which is more broadly based than the previous two, has been descibed by Wright (1993). A student's desire to be successful in games of chance could be exploited to evoke an interest in random numbers. A general problem which might be assigned would be to develop an automated method of picking four natural numbers between one and ten. This problem, which lends itself well to a spreadsheet-based solution, can be approached with varying degrees of sophistication according to the extent to which the solutions address the question of repeated number selection. Figure 7 shows two potential solutions.

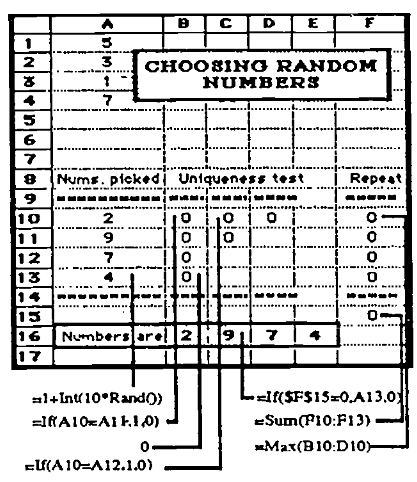

Figure 7.
Choosing four natural numbers between one and ten

The simpler of the two solutions is in cells 'A'1 to 'A4'. All this solution does is to employ the spreadsheet's random number generator to pick a number between one and ten—the prospect of selecting repeated numbers is not dealt with at all. The more sophisticated solution (in the block of cells 'A8' to 'G16') also does not avoid the selection of repeated numbers but it *does* check for their presence and will not print the selection in the dark-bordered box unless the four numbers *are* unique. Students may come up with one of many minor variations on the more sophisticated of the two solutions. An even more sophisticated solution might employ the use of macros to deal with repeated number selection.

Discussion

This paper has described three diverse problem solving contexts within which the spreadsheet might be employed to considerable advantage—many others could have been presented. It could be argued that the degree of spreadsheet competency required in the case of strategy 3 for the warthogs and cockatoos problem supplants the complexity of the traditional, analytical approach. It could be argued, however, that the spreadsheet approach offers two benefits, notably; that it is more visual (less abstract) and therefore easier to relate to, and that it reflects an appropriate use of technology in an information technology age.

In the "roots of equations" example, the spreadsheet can also be employed to great advantage by the teacher to demonstrate various properties of equations. In so doing, the teacher is provided with a valuable opportunity to role model the effective use of information technology (Wright, 1993).

While technology based tools such as Logo have been advocated for problem solvers at the elementary level (e.g.Maddux, 1989), spreadsheets have a great deal of potential for use at both the junior and senior high school levels Alhough not all problems will lend themselves to the use of the spreadsheet, such an approach *does* add a viable alternative to existing problem solving strategies. The greater the number of strategies available, the more likely it is that differences in learning style can be accommodated.

References

Maddux, C. D. (1989) Logo: Scientific Dedication or Religious Fanaticism in the 1990's, *Educational Technology*, 29(2), pp. 18-23.

Sheingold, K. (1991) Restructuring for Learning with Technology: The Potential for Synergy, *Phi Delta Kappan*, 73(1), pp. 17-27.

Wright, P. W. (1993) Teaching Teachers About Computers, *Journal of Information Technology for Teacher Education*, 2(1), 37-52.

Wright, P. W. (1993) Computer Education for Teachers: Advocacy is Admirable but Role Modeling Rules, *Technology and Teacher Education Annual*, 374-380.