TEACHERS MEET TECHNOLOGY:  
COMPUTER COURSEWARE AUTHORING IN  
SCHOOLS  
KENNETH A. SIROTNIK, University of Washington  
CLAUDE GOLDENBERG, Lennox School District  
JEANNIE OAKES, The Rand Corporation  

A look at the history of educational change suggests some realities about the future of computers in education. By and large, this history suggests a series of innovative “nonevents”—good ideas that never made it past the classroom door and the “cultural regularities” of schools and schooling that are often counterproductive to sustained educational change and school improvement. One clear lesson from this history is that those curricular innovations perceived as relevant and actually put into practice tend to involve the practitioners who use them in their development and adaptation. In studies of the successes and failures of attempted change efforts or collaborative research endeavors at the local school level, the same themes tend to emerge rather consistently. (1) teachers need to be conscious agents of their own change process, and (2) researchers need to enter into partnerships with teachers as they, together, explore the consequences of new ideas in the context of practice.

At first glance, the involvement of practitioners in the development of computer-related innovations may seem preposterous. School teachers and administrators, even those who are computer buffs, are unlikely to have the technical skill required for developing from scratch computer-based learning opportunities for their students. However, the availability of software with “friendly” authoring capabilities has made the prospect of locally developed or modified courseware less far-fetched.
The implications of this kind of curriculum development are far-reaching, especially when curriculum is broadly seen to include not only content but the entire way in which teachers organize and implement teaching and learning. In the process of authoring, for example, educators have the chance to explore the educational potential of computer technology in the larger context of the total educational experience for students. Teachers grappling with the process of courseware development are likely to gain a better understanding of what is possible with computers and to make careful and informed judgments about the roles they should play in classrooms. These opportunities will undoubtedly lead to more sophisticated conceptions of computer-assisted instruction than drill-and-practice exercises or games for enrichment that now dominate the market.

Further, teacher-courseware developers can focus on the needs of specific groups of students, develop a particular slant on content, utilize preferred learning activities and teaching methodologies, and incorporate local concerns and interests. They can avoid the sanitization and neutralization and, too often, the trivialization of subject matter and instructional approaches that often result when commercial producers attempt to be all things to all prospective buyers in a mass market. While many practitioner-developers may never develop the technical facility to produce all they conceive of as possible, they are likely to become both committed users and knowledgeable, demanding consumers in the marketplace.

Provoked by the potential benefits of involving practitioners in courseware authoring, yet mindful of the lessons learned from failed educational innovations, the Laboratory in School and Community Education (LSCE), Graduate School of Education, UCLA, undertook a year-long study of school-based courseware development. It became clear early in the Curriculum, Computers, and Collaboration (CCC) Project that, however great its potential, courseware authoring presents considerable problems and tensions for teacher-authors. It also seemed likely that these issues and how they were resolved would determine in large part how and whether courseware authoring would be used in schools.

The experiences of the project teachers illustrate what can happen when teachers meet computer technology—that is, when they, as courseware authors, confront both their perceptions of themselves vis-à-vis computers and the task of implementing their curricular intentions in this new medium. First, we present a summary of the CCC Project's main activities, followed by a description of the courseware authoring system. Against this backdrop, we then describe and interpret emerging issues as teachers encounter the system and attempt to translate their curricular ideas into computer courseware. Our

---

approach is basically one of case study by participant observation using both structured and unstructured sources of data—workshop and field notes, survey interviews and questionnaires, meeting minutes, anecdotal records, work samples, and the like. Our analysis of these materials is essentially reflective and qualitative, being based upon our interpretations of the context and processes of the project and the orientations (e.g., sentiments and opinions) of all participants.

THE CURRICULUM, COMPUTERS, AND COLLABORATION PROJECT

The CCC Project began in July 1983 as a collaborative effort between the LSCE and two public school districts. Three questions guided the inquiry. First, can participation in a collaborative development process lead teachers to conceptualize “ideal” curriculums? Second, based on conceptual work and practical experience, can the collaborative team develop curriculums that integrate microcomputers into classroom teaching and learning? Third, can the team use a courseware authoring system to design the elements of curriculums and implement their use in project schools? The scope of the investigation was limited to mathematics and language arts curriculums for students 10 to 14 years old.

The project team consisted of 15 individuals from four educational units, the LSCE at UCLA, and three schools. The computer expertise of the teachers varied enormously, from the computer buff to the “technophobic.” In between, some teachers had already had hands-on experience with micros and canned software; others had only seen some software demonstrated.

The project’s research and development activities included four related phases. The first was intensive conceptualization—planning, reading, thinking, and discussing curriculum ideas and the role of computers in education. The second phase consisted of developing tangible classroom strategies and materials based on the conceptual work. In the third phase participants received training on the authoring system, in order both to understand the technological constraints and considerations in developing computer materials and to gain facility in using the authoring system itself. The final phases included designing, storyboarding, and authoring courseware in mathematics and language arts.

THE COURSEWARE AUTHORING SYSTEM

LEADER is among the most comprehensive, state-of-the-art authoring systems we have seen. It is also, therefore, not the simplest of systems to use. Essentially, it includes a high-level programming language that allows the user to author lessons or courses fitting a generalized matrix structure. The matrix is composed of tasks, and each task is composed of a sequence of frames. The row and column structure of the matrix of tasks is open to the author's creativity. One typical structure uses task rows as levels of difficulty or discrimination, and task columns as levels of progress through the course content.

Built into the system is an elaborate response structure, which allows a variety of feedback and branching options based on student responses to questions posed in the frames of any task. Students can be branched forward in the matrix within the same row of tasks or to other rows or columns (e.g., different material of the same or different difficulty). It is also possible to branch students out of the entire course matrix and into a new one (e.g., for remediation, acceleration, elaboration, etc.).

Three kinds of files—text, graphic, and sound—can be created and edited. One or more segments from each can be selected and merged into a single frame (i.e., an image or sequence of images on the monitor screen). Frames can be built and edited, therefore, that have both text and graphics (and sound) in the same display. Graphics can be animated, and large amounts of text can be scrolled in separate parts of the same display. Assuming that the storyboarding and design work had been done, an experienced LEADER user could author a fairly sophisticated frame or sequence of frames, including text, graphics, and music, in several hours; the same software would likely take an expert programmer several months to do in BASIC.

---

We have reviewed a number of articles on courseware authoring, and we find no general agreement on a set of definitive characteristics distinguishing authoring systems from other systems. See, for example, Fred Bochman, "Creating Your Own Software with Mini-Authoring Systems," Electronic Learning 2 (1983): 72–75; Peter M. Dean, "Computer-Assisted Instruction Authoring Systems," Educational Technology 18 (April 1978): 20–23; Greg Kearsley, "Authoring Systems in Computer Based Education," Communications of the ACM 25 (July 1982): 429–437; Stephen K. Laver, "Maxi Authoring Languages in the Era of the Mini," Journal of Computer Based Instruction 4 (May 1977): 8–16; A. F. O'Neill and H. F. O'Neill, "Author Management Systems," in Instructional Systems Development, edited by Harold F. O'Neill (New York: Academic Press, 1979); and Richard E. Fogus, "The Authoring System: Interface Between Author and Computer," Journal of Research and Development 14 (Fall 1980): 57–68. Depending upon how general or specific you want to get, an authoring system is anything from a programming language (e.g., BASIC); to higher-level languages such as LOGO; to even higher-level languages deliberately structured around the architecture of educational lessons and courses (e.g., TICCIT by David M. Merrill, E. Schneider, and J. Fletcher, TICCIT, Educational Technology Publications, 1980); to the nearly language-free systems such as PILOT; to, finally, mini-authoring systems that require no programming skills at all—the user simply responds to requests for text (e.g., Shell Games for making up tests). LEADER is the name of the courseware authoring, student interactive, response management system developed by Edunetics, Ltd. The required hardware configuration is relatively simple: microcomputer workstations (e.g., Atari 400/800's plus color monitors), one or more disk drives, and one or more printers, all linked to a single hard disk (e.g., Corvus).
LEADER is a friendly system in that it requires no knowledge of programming. Nevertheless, friendliness is not synonymous with simplicity. LEADER requires a significant amount of training and practice to implement. Next we explore the successes and concerns that arose as a variety of teachers experienced the system in the CCC Project.

TENSIONS BETWEEN TEACHING AND TECHNOLOGY

As noted earlier, we are using the term curriculum in a broad sense, referring to all relevant elements of the teaching-learning experience: goals and objectives, content, instructional strategies, classroom activities, human and material resources, use of time and space, methods of assessment, and so forth. These curricular "commonplaces," filtered through an operant set of educational values and beliefs, by and large determine how teaching and learning is organized and conducted in schools and classrooms. These curricular dynamics shape what children experience and learn in schools. Students in one classroom, for example, may interact with the computer in largely routine ways, using drill-and-practice software and one-student/one-computer formats in computer laboratories unconnected to the organizing elements of class curriculums. In another classroom, computer experiences may be integrated with the purposes and strategies of instructional units involving small groups of students working together on problem-solving activities.

Early in the project we became aware of the subtle and not so subtle ways that the prospect of computerizing curricular ideas encroached upon the ideas themselves—and vice versa. The schematic in Figure 1 is intended merely as a heuristic for suggesting the kinds of tensions that can (and did) arise. The tensions represented by the reciprocal arrows can be briefly described as follows:

A = Seduction of curriculum inquiry and application by the specter of technology and vice versa.
B = Power of concepts and conceptualization versus the difficulty of reduction to operational terms.
C = Vision of unlimited potential of computer technology versus the state of the art.
D = Power of technology versus conceptual limitations.
E = Good ideas versus technological capabilities.
F = Attenuated concepts versus limited technology.


We, for example, would likely endorse such beliefs as equal access to curriculum regardless of race, sex, economic status, and so on, learning through critical thinking, discovery, experience, and so on, avoiding overreliance on only one pedagogical strategy, capitalizing on individual differences in peer tutoring, cooperative learning, and the like, and so forth. For further discussion of the interaction of the values/beliefs dimension with curriculum commonplaces, see Kenneth A. Sirotnik, "Evaluating Computer Courseware," Educational Leadership, 42 (April 1985): 39-42.
Essentially, the tensions arise between curriculum—as it is played out in the ordinary course of classroom practice—and computer technology—as it is played out in people's perceptions of its capabilities.

In the comments that follow, we look first at these tensions from the standpoint of technology and then of teaching. At the outset, we must emphasize that we see the tensions as not only inevitable, but as potentially productive when all individuals’ talents, skills, and dispositions are put to good use.

**The Technology Side**

Teachers authoring computer courseware must conceive of instructional interactions in ways they normally do not. They must anticipate student responses and carefully tailor courseware accordingly. The more meaningful and satis-
Understanding (from a teacher's standpoint) the instructional interaction, the more complex and formidable the task of conceiving and implementing it. The potential is enormous, but the process can go on interminably.

Teachers contend with questions and problems that arise during instruction on the run, so to speak, and with considerable facility. It is, however, quite another matter to anticipate and execute, far in advance and in a way that is educationally sound and constructive, several possible sequences of inputs and responses. The process may be likened to what chess players do when they play out, mentally, the likely outcomes of particular moves. Of course, as students engage in the interactions and teachers see what needs to be done, they can subsequently modify their computer lessons to make them more efficient, precise, or appropriate. But at least initially, teachers must plan the activities with only hypothetical students and student responses in mind.

Consequently, taking full advantage of an authoring system such as LEADER presents an obstacle of intellectual as well as technical dimensions. The problems LEADER presents are certainly partly technical: the system is complex and requires intensive training. Hence, the better the training, the easier teachers can be expected to employ the system. However, technical matters and training influence only one aspect of teachers' responses to being courseware authors; teachers must also reconceive how they plan, implement, and use curricular material.

The Teaching Side

Teachers generally teach lessons or plan activities in order to accomplish some more or less explicitly stated aim. To the extent that they feel their aims have been met, they feel satisfied and rewarded. To the extent that teachers feel their objectives have not been met, they feel dissatisfied and will look for alternative means for achieving their aims. Using traditional teaching strategies, teachers often can move their students satisfactorily along a course of study by articulating to themselves general goals and getting there in a way that is partly determined by unanticipated student responses. In fact, teachers expect unanticipated responses, and they rely on student questions and other indicators to signal when more explanation or review is needed. In other words, teachers do not articulate explicitly and in great detail how they will proceed in getting students to master a particular concept or skill. In their own interactions with students, teachers will make many spontaneous diagnoses and decisions that will determine the particular path that is to be taken in order to accomplish their educational aims.

Designing computer courseware requires teachers to function quite differently. They must be willing and able to re-cast their thinking about planning and designing learning activities. Teachers must confront the formidable task of carefully analyzing a concept or skill, constructing a meaningful and coherent approach to teaching it, and then designing a sequence of interactions between student and computer that will move the student toward accomplishing their teaching aims. Teachers must be willing to give up—at least in the
context of courseware authoring—their artful responses to unanticipated classroom events. Notions like seizing upon a teachable moment must be discarded in favor of an approach grounded in teaching as a science, where only students' predictable responses can lead to productive instructional interactions.

Teachers' reactions to this demand inherent in courseware authoring account, in large part, for the kinds of responses to be described later on. As we have said, the technical problems inherent in learning to use as complex a system as LEADER are certainly important for understanding teachers' responses to it. However, to ignore the way in which designing good interactive computer courseware forces teachers to alter how they think about, plan, design, and implement curriculum is to miss the very essence of what it means for teachers to be courseware authors.

In a sense, many of the teachers we worked with were victims of their own ambitious aims. They were committed to quite sophisticated educational objectives, such as having students understand how poets use sensory images to create moods, how writers develop characterization by means of various literary devices, or what the concept of fractions really means, rather than simply how they are multiplied and divided. Concepts that teachers could teach—and, indeed, had already taught—using traditional oral and written techniques, suddenly presented sizable problems when reconsidered for courseware development: How should the learning tasks be structured? In what sequence should subconcepts be presented? How many response opportunities and alternatives should students receive? What sort of feedback should they be given at various points in the lessons? Decisions that are routinely made during the course of planning and implementing learning activities were transformed by this new medium. And, again, while the technical obstacles were not trivial, the more substantive matters—questions at the heart of effective instruction and curriculum design—weighed heavily in determining teachers' responses to the authoring task.

There are several important points to keep in mind. First, when cast in the role of courseware authors, teachers must operate very differently from when using conventional approaches. Second, and intimately tied to the first, the obstacles teachers must overcome in order to become effective courseware authors are conceptual as well as technical. Teachers must be able to articulate explicitly the structure of the concepts or skills they wish to teach, they must design a coherent and meaningful sequence of interactions that will enable students to learn the concepts and acquire the skills, and they must relinquish the opportunity to create responses to students in the immediate process of their learning. A final point to remember is that if they cannot use the computer to achieve goals and objectives, teachers are unlikely to want to use it, no matter what potential an authoring system may have.

This final point will be especially important as we consider teachers' responses to being courseware authors. All of these responses can be productive and can lead to optimal use of an authoring system. Two key questions
then arise. Under what conditions are teachers likely to regard the authoring system as an aid to attaining their objectives? To what extent are they likely to become involved with the authoring system per se or with other activities related to courseware development? We will return to these questions in our final section on the implications of our findings for the future of courseware authoring systems in schools.

TEACHER RESPONSES TO COURSEWARE AUTHORING

In our work over the year, we noted a variety of ways teachers tend to respond when asked to function as courseware authors. We categorized their responses as withdrawal, computer-driven, instructional technologist, and actualization of the computer medium. The lines separating these response types are somewhat blurred, and any one person may have exhibited more than one type of response as time went by and conditions changed. However, an awareness of these response types may be useful in attempting to anticipate how teachers will react when asked to work in ways far different from how they may normally function.

Withdrawal

Many teachers, in particular just after the initial training, abstained from developing any sort of courseware, or even from attempting to sketch preliminary outlines of possible computerized activities. There were three reasons for some teachers withdrawing. One problem was technical. They felt incompetent and intimidated by the system. Training was too short. Teachers did not have sufficient time to become comfortable with the system. "I was a miserable failure at programming," wrote one teacher on a year-end questionnaire. Second, some teachers were at a loss as to how they should proceed conceptually. How does one go from an initial conception for a lesson or a group of lessons to a programmed set of instructional interactions? The task seemed so overwhelming (even before one had to worry about programming specific commands), and the day-to-day pressures of teaching and dealing with administrative demands were so immediate, that the development and implementation of computerized lessons was relegated to a low priority. These first two obstacles were, in part, overcome by providing teachers with technical assistance and by recommending concrete steps for moving from a specific instructional idea to computer implementation.

The third obstacle, however, was more difficult to overcome. Some teachers felt that the system was not well suited to the higher-level cognitive activities fundamental to good curriculum and instruction. There was some disagreement as to whether such activities were impossible to computerize, or whether they were possible in principle, but extraordinarily difficult in practice, requiring a high level of technical expertise and considerable time and effort. In either case, these teachers believed that their instructional objectives could be better met with more traditional instructional technolo-
gies. One teacher (who initially was quite favorably disposed toward learning and using the system) indicated that LEADER seemed suited to activities that require students simply to demonstrate recall or comprehension skills. The system might also be useful, noted the teacher, for curriculums involving higher-level thinking (such as application and analysis of information), but only if the questions and answers were "very predictable." Combined with the formidable technical task of learning how to maneuver within the system, even at a minimal level, this perceived view of the system's inherent educational limitations led some teachers to conclude that their instructional goals could not be well served by devoting large amounts of time to courseware authoring.

To summarize this rather heterogeneous category of teacher responses, teacher withdrawals occurred for a variety of reasons ranging from (1) initial technophobia, to (2) frustration with mastering a complex system with so many other demands on time, to (3) the difficulties of putting it all together—translating curricular concepts into the kinds of lesson and course plans that fit the structure of the authoring system and then entering those plans into the system using the appropriate command language. Regardless of their degree of withdrawal from the system, we think it fair to say that all teachers felt positive about the system's potential. Given unlimited time, relief from other professional activities, and technical support, it is likely that all could have become highly involved in courseware development.

**Computer-Driven**

A second way teachers responded to courseware authoring was for them to concentrate on the technical aspects of the authoring system to the neglect of courseware development. This response generally took one of two forms. In some cases, teachers became totally engrossed in trying to master basic elements of the system to the exclusion of courseware development. These teachers, as was true of many who had withdrawn, completed the training without feeling that they had a good grip on the system. Unlike those who withdrew, these individuals devoted much of their project time to mastering the exercises provided during training, leaving almost no time (unless it was their own) for courseware development and authoring.

A second form of the computer-driven response was for teachers to design very simple, or simplified, interactions that they felt they could program easily. In this case the form, and in some cases the substance, of the lessons became dictated by what seemed to be most immediately feasible and could be accomplished relatively quickly. This necessarily involved sacrificing elements of a lesson that would make it more instructionally sound or interesting, but also more complex to design and program. An example of such simplification is giving generic positive ("right") and negative ("wrong") feedback messages rather than designing specific feedback for particular anticipated correct and incorrect responses.
In some ways, these two forms of the computer-driven response are opposite sides of a coin. In the first one, teachers had a genuine interest in mastering the system (or some manageable chunk of it), and they devoted almost all allotted time to it. In the second form, teachers had much less interest in mastering the system, they simply wished to construct and author lessons that seemed to be most compatible with the system and their skill levels in working with the system. Both forms, however, demonstrate how technology can influence both the quality and quantity of courseware production.

**Instructional Technologist**

A third category of response included teachers who attempted both to master the system sufficiently well (or, alternatively, who made use of technical support from other project personnel) and to develop courseware that would accomplish specific instructional goals. These teachers were less computer-driven than those previously described in the sense that they were concerned about using the computer for accomplishing particular instructional objectives. Although the means for accomplishing them were often not immediately apparent, individuals whose response could be characterized as "instructional technologist" were willing to put forth considerable effort to overcome obstacles and difficulties. Teachers whose responses fell into this category were much less likely to compromise their instructional aims, even if abiding by them meant more planning, designing, and programming. Teachers in this category were most concerned with developing high-quality, self-contained computer courseware, and their efforts were directed toward this end.

**Actualization**

The final category of teacher response was the rarest but we think the one most likely to exploit fully the computer's potential as an instructional tool. Essentially, teachers whose responses fell into this category used the computer to accomplish specific aims when doing so was judged to be appropriate, but employed other media when they seemed more appropriate. The computer was not used for its own sake, nor was its use necessarily an end in itself. As one of the teachers wrote, the computer is "another aid to learning."

A good example of this approach came toward the end of the year. Two teachers, working together, wanted a final evaluative activity wherein students were to write several lines of poetry to demonstrate their mastery of a particular concept. Originally they had intended to have students type the lines on the computer using a writing pad developed by another teacher in our project. However, because what students wrote on the writing pad could not be saved, the teachers would have to go from computer to computer before they were turned off in order to read students' writing. This was clearly impractical, so the teachers were faced with a choice: either change the evaluative activity to one that is more amenable to the computer, or have the students use more traditional technologies for the evaluation—in this case, pencil and paper.
The teachers opted for the latter; it was important for their purposes that students be able to demonstrate, in writing, their understanding and application of the concepts they had been taught. It was also important that the teachers be able to read and evaluate what students wrote. The teachers developed interactive courseware to present the material and to conduct interim evaluations of student learning. Final evaluation, however, required a different medium.

Obviously, the authoring system's inability to save written student input is a technical problem that can be overcome by making certain changes in the software, thus expanding the system's capabilities and eliminating this particular dilemma. However, there will always be limitations to what computers can do (at least until the promises of artificial intelligence are realized, although even this is being called into question*). An important element of the computer's instructional value is knowing when using it is appropriate and when it is not.

In order for teachers to respond in this way to courseware authoring, two conditions seemed necessary. First, teachers must have sufficient technical expertise—or at least support—to permit them to (1) design the computerized portions of the lessons, and (2) recognize the practical limits of the authoring system. Second, teachers had to have explicit goals for their lessons and instructional activities, and their primary aim had to be accomplishing these goals. By knowing specifically what they wished to accomplish and why, teachers seemed able to prevent their educational objectives from becoming overwhelmed by the technology and the allure of the computer. Only then could they view the computer in the appropriate context—as a tool for reaching instructional aims, not a device that defined or limited them.

**IMPLICATIONS: USING AUTHORING SYSTEMS IN SCHOOLS**

Obviously, people respond in different ways to deliberate changes in their daily work life no matter how smooth or rocky the intervention seems to be. To assume that people within an organization will react similarly to innovative activities would be to ignore a fundamental premise of psychology—individual differences. To assume that what occurs in one organization can be replicated in another ignores a basic premise in sociology—group differences.

What we have reported here is simply one set of results and interpretations based on what we believe to be a moderately successful attempt to involve teachers in a process almost 180 degrees opposite from that to which they are normally accustomed: a process that allows opportunities to reflect on and discuss what ought to constitute the curriculum in a particular subject field, to translate these concepts into course and lesson plans, and to translate these plans into computer courseware. We attempted to do this not in a

laboratory setting, but within the exigencies of teachers' daily work lives in schools. In this context, for example, time is an extremely scarce commodity. Yet it is the most important resource for adequately carrying out all the phases of a project of this nature.

The point is that just as various features of the Curriculum, Computers, and Collaboration Project were less than ideal, so are the circumstances of schooling and the state-of-the-art of this or that authoring system. The issue then becomes not how to avoid certain teacher responses so much as how to anticipate and capitalize constructively on the various responses. Some people do not want to become (or at least develop the skills of) computer programmers, which is what they need to do to become quality courseware authors using a system of LEADER's sophistication and capability. Other people love the challenge of technology and the work necessary to translate curricular storyboards into the computer medium. Not only is there room for these (and other) teacher types in an authoring environment, but it is probably necessary to develop a multi-skilled approach to school-based authoring, given the daily pressures of the setting.

Consider our project results as a case in point. All teachers whom we might have classified as withdrawals were eager to remain involved in curriculum inquiry and development. Some teachers preferred not to spend the time necessary to master the system. Others simply preferred to concentrate on curriculum design, leaving the courseware authoring to those more inclined toward programming.

On the other side of the coin are the computer buffs who are either already well versed in the technology or who become so in a relatively short period of time. Interestingly, there was no apparent correlation between technical facility and type of teacher response. Some who withdrew had previous programming experience. Some who began to actualize courseware development as an integral part of the larger curriculum spectrum and/or behave as an instruction technologist required, in fact, considerable technical support from others who were more facile with the authoring system.

Perhaps the only kind of teacher outcome experienced in this project that we would consider "negative" is the computer-driven response that takes the easy way out in tailoring the curriculum to what is implemented most expeditiously with the authoring system. Fortunately, this is the response most easily avoided and redirected when courseware authoring is viewed not as a mandatory skill for all, but as a cooperative curriculum development task requiring varied talents, dispositions, and attitudes.

Although risky to generalize from our single experience in the CCC Project, we can at least speculate on what the generic features might be for developing this kind of multi-talented/multi-interest approach to courseware authoring by teachers in schools.

1. Teacher involvement should be voluntary. Any change process based around mandatory participation is destined for failure.
2. All participating teachers need to become familiar enough with computers and the authoring system to (a) overcome "technophobia," (b) demystify technology and microcomputers, and (c) understand the potential and limitations of the authoring system itself. All teachers can attain an adequate understanding of a system like LEADER by participating in a well-designed (consisting of user-friendly manuals, carefully prepared exercises, appropriate opportunities for exploratory learning, etc.) two-week, concentrated training program.

3. All participants need to engage in critical discourse around fundamental curricular issues underlying the various decisions teachers make when operationalizing concepts into course outlines and lesson plans. For example, is mathematics to be viewed primarily as operations, percentages, word problems, and the like; or as problem solving, abstracting, proving, and so on, thus building in computational skills along the way? These alternative views can have important consequences for how curricular ideas are ultimately translated into courseware.

4. Individual talents and interests must be encouraged and nurtured in curriculum inquiry, design, and courseware development. The purpose of items 2 and 3 is to provide all teachers with a sufficient grounding in the curricular and technological components of courseware development and authoring so that they can knowledgeably and productively pursue their special contributions. At minimum, each school site would need (a) one or more persons with considerable software and hardware expertise to serve as computer resource persons, (b) one or more persons with curriculum design and storyboard production skills in each of the content areas under consideration, and (c) one or more persons who are curriculum generalists (i.e., who can step back and constructively analyze courseware development in the larger curricular context). These abilities need not each be housed in separate people, but they need to be actively represented in the group of participating teachers.

CONCLUDING REMARKS

There is much more to be done and said about the interface of teachers and courseware authoring systems. For example, the issue of courseware quality and the evaluation of courseware, particularly as it is introduced into the classroom learning environment, has yet to be studied systematically. More rigorous study on training teachers as authors (dealing with individual differences, for example) is also needed. But most important, no matter what variations may take place on the generic features enumerated above, district and school administrators and teachers must realize the necessity of, and make provisions for, meaningful released time for participating staff members. To do so will necessarily require reconfigurations of how teachers ordinarily conduct their school day. Taking courseware authoring systems seriously in schools will require an equally serious commitment to making them work.
To return to the theme that opened this report:

Computers are capable of profoundly affecting science by stretching human reason and intuition, much as telescopes or microscopes extended human vision. I suspect that the ultimate effects of this stretching will be as far-reaching as the effects of the invention of writing. Whether the product is truth or nonsense, however, will depend more on the user than on the tool.9

Anthony Oettinger made these statements in 1969 when a 128K, second generation computer filled up the space of a large classroom and cost $11 million. Now a computer with four times the memory can fit on the corner of your desk for 1/400th of the cost. Yet Oettinger's words are as relevant today as they were 17 years ago.

If the history of educational change has any predictive validity, the use of microcomputers in the classroom may simply model old classroom conventions and configurations in a new medium. However, if educational researchers and practitioners have the foresight and necessary support, we may see an exception to this rather dismal rule.

KENNETH A. SIROTNIK is Research Professor, College of Education, DQ-12, University of Washington, Seattle, Washington 98195.

CLAUDE GOLDENBERG is Teacher, Lennox School District, 10319 Firmona Avenue, Lennox, California 90304.

JEANNIE OAKES is Social Scientist, Rand Corporation, 1700 Main Street, Santa Monica, California 90406.


Proceedings of the Sixth Invitational Conference (1984) of the Canadian Association for Curriculum Studies. Seven major papers with responses Connelly and Clardinin (on six perspectives on inquiry), Barrow (on reorienting curriculum research), Egan (on developing sensitivity to forms of understanding), van Manen (on a theory of the unique), Landry and Robichand (on processus enlignement-apprentissage), Beattie (on computer knowledge), and Beebe (on vocational education).

---
