

COMPUTER-SUPPORTED INTENTIONAL LEARNING ENVIRONMENTS

MARLENE SCARDAMALIA

CARL BEREITER

ROBERT S. McLEAN

JONATHAN SWALLOW

EARL WOODRUFF

*Centre for Applied Cognitive Science
Ontario Institute for Studies in Education*

ABSTRACT

CSILE, which stands for Computer-Supported Intentional Learning Environments, is an educational knowledge media system. CSILE allows information in several media (text, drawings, graphs, timelines, etc.) to be entered into a common database where it is available to be retrieved, linked, commented on, rated, and so forth. The environments and operations of CSILE are designed of information. In this article eleven principles, based on recent cognitive research, are suggested for designing computer environments that support intentional learning. These principles include making knowledge-construction activities overt, maintaining attention to learning goals as opposed to other goals of an activity, providing process-relevant feedback, and giving students responsibility for contributing to each other's learning. Applications of these principles in CSILE are described, as well as observations from the first year of school try-out.

BACKGROUND

There has been a history of attempts in computer-assisted instruction to give students more autonomy or more control over the course of instruction. Usually these attempts have had disappointing results [1]. But these attempts *presupposed* a well developed repertoire of learning strategies, skills, and goals, without providing means to foster them. When CAI emerged in the 1960s there was not the scientific basis that would have made it possible to design programs that actually helped students learn how to learn, learn how to set cognitive goals, learn how to apply effective strategies for comprehension, self-monitoring, and organization of knowledge. Cognitive research of the last fifteen years, however, has made substantial progress in providing a basis for such programs.

The purpose of this article is to suggest principles for applying these recent findings in designing computer environments, to describe an implementation of these principles (called CSILE), and to report observations from the first year's trials of this implementation. We use the expression

computer-supported intentional learning environments to refer generically to environments that foster rather than presuppose the ability of students to exert intentional control over their own learning [2]. We use the acronym CSILE to refer to the particular computer-supported intentional learning environment currently under development in our laboratory.

CSILE supports intentional learning by providing a means for a group of students to build a collective database (knowledge-base) of their thoughts, in the form of pictures and written notes. CSILE stores the thoughts entered by each student and makes them available to everyone. Students use a color graphics editor to create their pictures. CSILE encourages students to organize these pictures to allow “zooming in” to a blow up of a section of a picture or “zooming out” to see a broader picture. Written notes can be labelled in a variety of ways. Students are asked to provide these labels in order to facilitate reflection and to allow the notes to reappear in multiple contexts. In addition, written notes can be placed on a timeline, or attached to a spot on a picture. The system is a form of hypermedia that allows notes entered as text, drawings, graphs and timelines to be retrieved, linked, commented on, rated, and so forth.

CSILE was initially developed for university and graduate level students. Results of trials with early versions showed that students were being encouraged to think more about how they process and reprocess thoughts on research literature and class projects. The first-year implementation of CSILE focused on two grade five-six classes. There were a total of sixty-four students using CSILE three or more times each week. CSILE is being made available on sixteen *networked* ICON microcomputers, eight in each classroom.

CSILE is being designed for eventual use at all grade levels and for all school curricula. It will also act as an umbrella over a variety of computer-based and conventional school activities, providing a second layer of problem solving and learning activity.

One of the objectives of the CSILE project is to develop specifications that are applicable to a wide range of educational software and that are concerned with increasing the ability of the software to support reflection, problem solving, and learning. One reason such specifications are needed is that there is a potential conflict between the principles that inform most software development and those that ought to guide development of educational software. In most software design it is presumed desirable to make the software as intelligent as possible and to demand as little intelligence as possible from the user. Educational applications on the other hand, should be aimed at developing the intelligence of the user. Educationally irrelevant burdens should be minimized, but not in ways that deprive students of occasions to develop the planning, monitoring, goal-setting, problem-solving and other higher-order abilities that are important objectives of education.

RESEARCH ON HIGHER-ORDER ABILITIES IN THINKING AND LEARNING

Unlike earlier research, which was concerned with the *correlates* of academic performance [3] recent cognitively oriented research is concerned with what students actually *know* and *do* that makes them relatively successful or unsuccessful. (A book edited by Chipman, Segal, and Glaser [4] contains reports by many of the contributors to this rapidly growing body of research.) Successful learners are found to use a variety of cognitive strategies and self-management procedures to pursue knowledge related goals, to relate new knowledge to old, to monitor their understanding, to infer unstated information, and to review, reorganize, and reconsider their knowledge. These active learning processes are what need to be supported by computer-supported intentional learning environments.

Of equal importance to the design of computer-supported intentional learning environments, however, is an understanding of the strategies that characterize more passive or immature learners. These characteristics include 1) the organization of mental activity around topics rather than goals, 2) a focus on surface features. 3) the use of straight ahead rather than recursive procedures, and 4) an additive rather than a transformational approach to learning. These characteristics have been identified in a variety of curricular areas, including mathematics [5, 6], science [7], reading [8-10], and writing [11]. There is also evidence that these passive learning characteristics are implicated in specific learning disabilities [12].

A very important finding related to these immature strategies is that they have great persistence. They tend to be at least partially successful in school situations, and so students tend to cling to them even though they are quite inadequate for real world learning needs [13]. However, there have been a number of successful experiments in teaching more active learning strategies [9, 14-16]. The successful teaching efforts generally include a combination of modeling and explanation of strategies, along with some form of support and encouragement of students' taking over greater responsibility for the learning process. It seems clear from these findings that in order for a computer system to provide maximal support for the development of higher-order abilities, it will have to encourage active rather than passive learning strategies and give students help in sustaining the more active approaches to learning.

The key question is how such support can be provided. At first thought it might seem that the key requirement is a highly intelligent tutoring system, capable of performing sophisticated diagnoses of students' learning strategies, providing sensitive coaching and instruction. Such a system, however, is not only unrealistic, given the state of the art; it may also be heading in the wrong direction [17-18]. For it is not the computer that should be doing the diagnosing, the goal-setting, and the planning, it is the student. The computer environment should not be providing the knowledge and intelligence to guide learning, it should be providing the facilitating structure and tools that enable students to make maximum use of their own intelligence and knowledge. That is the idea behind *procedural facilitation*.

Procedural facilitation [11, 19, 20] is an instructional approach that grew out of efforts to foster higher-order processes in written composition. It is a

theory based approach to providing learners with temporary supports while they are trying to adopt more complex strategies. These supports include turning normally covert processes into overt procedures; reducing potentially infinite sets of choices to limited, developmentally appropriate sets; providing aids to memory; and structuring procedures so as to make it easier to escape from habitual patterns. A cardinal principle is that these supports should be designed so that when they are withdrawn the learner is carrying out the mature process independently. Thus it is essential that design of procedural facilitations be based on adequate models of the mature process, and that it also be informed by models of the immature process.

As Brown and Palinscar have pointed out [21], procedural facilitation is akin to other kinds of support for learning, such as the support parents provide for young children in the initial stages of mastering conversational discourse. However, procedural facilitation has the distinctive advantage of being applicable to computer-mediated learning. Most other kinds of support for learning require mother/child interactions described by developmental linguists [22] require a degree of sensitivity far beyond that of the most intelligent of computer-based tutoring systems. In procedural facilitation, however, the intelligence is all in the learner. The supportive system need only be able to do the kinds of things that computer systems are already well equipped to do—presenting knowledge—structuring environments, presenting formats and reminders, storing and retrieving information, facilitating (through menus, for instance) the making of choices, and monitoring the sequence of events. Thus the development of procedural facilitation would appear to be crucial to making computer-supported intentional learning environments a possibility.

DESIGN PRINCIPLES FOR COMPUTER-SUPPORTED INTENTIONAL LEARNING ENVIRONMENTS

The following specifications grow out of the background research, planning, and initial trials of CSILE. They should be regarded as provisional, to be revised and augmented as CSILE development and formative evaluation proceed. Each specification is presented first in terms applicable to a variety of educational software: then the particular application of the principle to the development of CSILE is described. This description includes examples of how the principle is being realized.

Make Knowledge-Construction Activities Overt

Most of what intentional learners do is hidden from view, consisting of goal setting, identifying and solving problems of understanding, connecting old and new knowledge, and so on. Wherever possible, these activities should be made overt and identifiable, so that students become aware of them and become better able to carry them out deliberately. Instead of only providing menus of topics and tasks, for instance, the computer might provide menus from which

TABLE 1 Examples of Notes Labelled via the Student-Designed
“Confusion” and “New Learning” Icons

I don't understand how bacteria can help us digest our food. They're too small to have a mouth.

In a lake, why doesn't the water just go through the sand and doesn't leak or anything?

Do germs in an African person resemble germs in an Antarctic person?

New Learning

Charles Darwin was born in 1809 in the town of Shrewsbury in England on the same day as Abraham Lincoln.

It's true that once in a blue moon the moon is really blue. The last time the moon was blue was 1950. Scientists believe that it was caused by a thicker coating of filtered dust that filtered out red light.

I have learned many things by reading this article....I would have thought that your eye movement was controlled by the brain. One of the convincing parts of the argument is when the eye is patched, the brain would not know the eye position was changed but the eye did respond and that meant that the muscle sent messages to the brain

students select the kind of mental activity they intend to engage in. as much as possible, these overt mental activities should have consequences within the functioning of the software so that they are not merely additional burdens placed on the user. For instance, choice of a goal could determine which of several specific choice menus is presented; discoveries or hypotheses could go into a file that plays a role in other users' progress through the same game or microworld.

CSILE application - Student-designed icons representing thinking types are used as one way of identifying notes for subsequent retrieval by self and others. Notes often clearly reflect the student's selection of thinking type. Examples of notes labelled via the student designed “confusion” and “new learning” icons (to identify ideas they find confusing or novel, respectively) are presented in Table 1. In some cases patterns can be seen where students begin with some new learning, generate questions, and then produce plans for the investigation of those questions.

Maintain Attention to Cognitive Goals

Students should be called on to state their goals, to anticipate what they will learn and what they will do en route to attaining their goal, and possibly to

specify a time at which they think they will have reached their goal. As much as possible these should be *cognitive* goals (learning, finding out, etc.) rather than *task* goals (such as scoring a certain number of points, finding the treasure, etc.). A course project by R. Lloyd compared an expert and student driver using a driving simulator. The expert focused on goals relevant to real world driving (e.g., obtaining information, anticipating problems, informing other motorists of one's intentions) whereas the novice explicitly treated the simulation as an arcade game and focused on game-related goals. Well-designed educational software should encourage the type of goal orientation shown by the expert rather than that shown by the novice.

CSILE application - Planning is one of the thinking types selected with the help of student designed icons. It will be supported by prompts to indicate cognitive goals, plans for pursuing them, and target dates. For example, the following was entered by one child.

Goal: To Use the Timeline by February 19

1. Find out when World War 2 started and ended
2. Find out about 1920s.
3. Find out the date that William the Conqueror conquered England: may be look under "The Tower of London."

Students have begun to use the timeline facility to schedule deadlines for following up on questions they've chosen to investigate. However, novice abilities become evident when plans are not further addressed. In an attempt to resolve this problem, we are planning to implement a calendar function that will provide an overview of learning goals previously established by the student and check for progress on goals scheduled for the current log-in date.

Treat Knowledge Lacks in a Positive Way

Typical school activities provide opportunities to display knowledge, but knowledge lacks usually emerge as forms of failure. Yet knowing what one does not know is a vital kind of metaknowledge, without which intentional learning is severely limited. Wherever possible, educational software should provide means for students to identify what they don't know or need to find out or are curious about. Wherever possible the accurate identification of knowledge lacks should have positive consequences within the functioning of the program, resulting in enhanced possibilities for achieving the goals that are motivating use of the program in the first place. Identified knowledge lacks can also serve as valuable material for analysis, both within computer-assisted activities and in related classroom discussion.

CSILE application - Raising questions reflecting interests or curiosity about material they are studying is one of the main things students do in CSILE notes. Class activities, procedural facilitations, and feedback provisions within CSILE are used to encourage serious question formulation and to help students

upgrade the quality of their questions. Already students have demonstrated their willingness and ability to express what they are curious about, as indicated by the brief sampling of their notes in Table 1. However, some students may still be unwilling to share their uncertainty with others, and they either store their notes anonymously, or they choose not to let CSILE pass their notes on to others. Related research is seeking to identify ways in which students can pursue matters of interest through peer interactions that engage all students.

Provide Process-Relevant Feedback

Although there is much talk at present about focusing on processes rather than products in education, this is often difficult to do with educational software, just as it is in ordinary classroom conditions. Intelligent tutoring systems aim to provide such feedback. A possibly more fruitful educational approach, however, is to design partner or team activities in which one member has the job of monitoring processes and is provided with computer support for doing so-in the form of cues, menus, recording formats, etc. side-by-side computers running parallel programs provide one way to do this: simultaneously active windows provide another.

CSILE application - In its present implementation CSILE provides process-relevant feedback only indirectly: Teachers can obtain reports of the number and kinds of notes produced by each student, including the thinking types chosen, and teachers can then provide feedback to students as they see fit. In

Table 2. Correspondence between Children’s Self-Reports and Independent Assessments of Level of Mental Effort

<i>Self Reports</i>	<i>Level of Mental Effort</i>
“They all said the same things in different ways. I didn’t learn anything new.”	1
“I just do not have enough time so I copied some other notes into my files.”	2
“I didn’t get any new ideas but I felt my ideas were on the right track as the other notes confirmed and supported my ideas.”	3.5
“I included these notes because the authors seemed to more clearly express the role of the palisade endings in describing how muscles send messages to the brain.”	3.5

Notes: Mental effort ratings are on a scale of 1 to 5.

future versions, however, students will be able to obtain reports directly. In addition, there will be provision for a variety of self-ratings and ratings by others (on how much effort they think went into a note, on how helpful they found it, and so on), and the system will be able to summarize these data for the user. Students' self-ratings have demonstrated surprising insight. Table 2 presents students' justifications for ratings of the amount of mental effort they exerted during a classroom exercise that required them to review peer notes and construct a summary.

Independent assessments suggest that student self assessments conform to assessments by readers who had not seen the student's own rating. Supportive contexts for self-evaluation will, we believe, allow us to create increasingly effective conditions for process-relevant feedback. Another route for process-relevant feedback currently being investigated supports partnership activities using a parallel program. A split screen is used to provide a limited set of CSILE facilities for a student to create notes, while at the same time providing support to a second student who is giving feedback and support to the first student. The supporting students provide feedback about the clarity, consistency and completeness of their partners' verbalizations. In addition, the supporting students help to focus their partners' thinking efforts toward the examination of relevant prior knowledge.

Encourage Learning Strategies other than Rehearsal

Less successful students tend to rely on rehearsal and memorization strategies, whereas more successful ones supplement these with strategies aimed at understanding [13]. Drill-and-practice approaches in educational software naturally

Figure 1. Predictions of grade six students regarding passage of messages between eye and brain.

emphasize rehearsal. These should be supplemented with approaches that emphasize comprehension strategies. This does not mean just asking comprehension questions. There are other approaches that use machine-interpretable data that elicit even more active use of comprehension strategies. Examples are arrangement tasks (arranging the sentences in a scrambled text, lines of program code, steps in a proof, etc.) and cloze procedures using paraphrases rather than copies of material students have previously studied.

CSILE application - CSILE emphasizes understanding related strategies by calling on students to identify and deal with confusions, problems, insights, and critical judgments of information. Thus, for example, we see students reviewing notes that they find in the database and compiling them under a heading such as "Questions that I must answer." We also see use of prediction. Figure 1 shows three notes and accompanying illustrations that represent different students' hypotheses about the passage of messages between eyes and brain. The issue of how messages travel arose in the course of grade five to six students' reading a difficult text about feedback from eye muscles to brain. They posed questions that peers tried to answer, and this prediction making activity was supported by the interactive environments and diagramming facilities of the system. On the other hand, rehearsal will not be neglected. Planned enhancements of the system enable students to schedule and create self-tests and reviews and to cooperate in testing one another.

Encourage Multiple passes through Information

In contrast to experts, naive students show a strong tendency not to go back over information. This is shown in their reluctance to revise compositions and to check work in mathematics. Deeper experimental analyses have shown naive reading and writing to be characterized by single-pass strategies, in contrast to the recursive strategies of experts. Because computers make it easy to retain and recall information, educational software has the potential to help students develop multiple-pass strategies. Whether the software involves games, microworlds, subject matter learning, or tool use, ways should be sought to make it worthwhile for students to call back information they have dealt with previously and to reconsider it or to use it in a different context.

CSILE application - CSILE provides a variety of occasions for multiple passes through information. Through database searches using different search criteria, the same note may appear in different contexts and in different knowledge structuring environments (for example, it might appear on a timeline or on one or more charts). Scratchpad, note editing, and copying functions encourage students to reprocess previously recorded information. Experience to date suggests, however, that if students are to reprocess information to any great extent, school assignments have to be structured in such a way as to encourage it. A particularly powerful use of CSILE occurs when teachers assign a series of activities that require students to call up information from sources they have used before and employ it in increasingly sophisticated ways. For instance, at one point students record new things they have learned from a unit of study. At another point they must summarize the most important ideas of the unit, which involves them in reviewing other students' notes on new learning. At a still later point they must formulate questions for further study, which involves them in going back over the previously produced notes and summaries to look for gaps, unanswered questions, and ideas worth pursuing. Student comments reported in Table 2 show students' self-ratings during one such series of activities.

Support Varied Ways for Students to Organize their Knowledge

This principle is stated in very general terms, because its application will vary greatly depending on the nature of the educational software. The general idea is that the easiest way to organize data via computer is in the form of hierarchical lists (as is done with directories and subdirectories, for instance). Thus it is natural, whenever students are required to deal with data in an organized fashion, to present it and have them store it categorically. This method of convenience is sometimes rationalized on the grounds that it teaches “categorization skills” or something of that sort. The trouble is that for children of age eight or beyond categorization is already their natural way of organizing knowledge and what they need most is help in growing beyond that limited structure. Immature learners, for instance, tend to store knowledge as discrete details organized under topical headings and therefore fail to grasp arguments, lines of thought, or major themes [10]. There are many alternative ways of structuring knowledge, such as timelines, graphs, maps, narrative sequences, story grammar structures [23], concept nets [24], and causal chains. The suggested principle, therefore, is that whenever software developers are about to involve students in working with a hierarchical list structure they stop and give thought to the possibility of using some other way of organizing information.

CSILE application - One of the most distinctive features of CSILE is the provision of different ways of representing knowledge, all of which are accessible in the same database. Currently implemented alternatives include maps and other diagrams that are hierarchically embedded so that students can zoom in or out to different levels of detail and attach notes at any level, and provision for freely constructed pictorial as well as written notes, and a timeline to which students can attach notes. For example, students used the “zoom-in” feature of charts to illustrate what they saw as they dissected a frog (see Figure 2). Their notes were attached to and organized by these charts, so that, for instance, a note about the frog’s heart was accessible by taking the cursor to the letter over the diagram of the heart and pressing the action key.

Figure 2. A grade six student’s use of CSILE’s zoom-feature to illustrate (I) “External features,” (II) “veins/arteries,” and (III) “organs” as she saw them while dissecting a frog.

Encourage Maximum Use and Examination of Existing Knowledge

A serious shortcoming of most educational software is that the software itself has access to a very limited body of knowledge (e.g., rules of the game, results of arithmetic operations, answers to stored questions) and the student must work within the confines of that limited knowledge base. Real world tasks, on the other hand, tend to be wide open in the kinds of knowledge that may be drawn on in handling the task. Educational software that draws on large databases is one answer, of course. A less costly alternative is to use the computer to support more open-ended tasks, in which students may draw on knowledge from a variety of sources (including their own world knowledge), and where feedback is not provided by the computer but by the teacher or other students.

CSILE application - CSILE pursues the less costly alternative, but in a way that has many of the advantages of large knowledge-based systems. CSILE does work with a large knowledge base, but the knowledge base is constructed by the students. The students are not only responsible for putting knowledge into the system, they are also responsible for evaluating it, interrelating it, labelling and sorting it, and performing periodic reorganizations and house cleanings to enhance the quality of the community knowledge base. Thus the students get experience in many aspects of working with large knowledge bases. In later versions of CSILE, prepackaged databases and video technology will be incorporated.

Provide Opportunities for Reflectivity and Individual Learning Styles

One of the oft cited advantages of computers in the classroom is that they provide an alternative route to learning, which may be especially valuable for students whose personal learning styles are not suited to the pace and publicness and classroom learning. In order to support reflective thinking, however, educational software must do more than permit private, self-paced learning. The program must provide students time, opportunity, and peace in which to think about what they are doing and why. This means the program should not be so busy "motivating" the learner that it keeps up a bombardment of stimulation, and it should not be so structured that it is always controlling what the student thinks about.

CSILE application - CSILE is not so much an alternative to classroom learning as it is an environment for quiet, reflective thinking that precedes class discussion. It provides (if student so chooses) anonymity in posing and answering questions and private, computer-managed interchanges during which students can reflect on ideas based on their own and other students' responses. CSILE provides a scratch pad for making rough notes, a facility for placing notes on hold while other ideas are pursued, and a feature which allows notes already in the database to be edited.

Facilitate Transfer of Knowledge across Contexts

The compartmentalization of learning into school subject areas has long been blamed for students' failures to transfer knowledge to new contexts [25] but other curricular considerations make it difficult to eliminate compartmentalization. Educational software has an opportunity, not enjoyed by textbooks, to cut across curricular lines. Computer microworlds, for instance, may be designed to apply knowledge from several disciplines-physical science and economics, for instance; or biology and geography. Games, similarly, may be designed to cross curriculum boundaries.

CSILE application - The databases of insights, problems, goals, etc., that CSILE compiles are accessible to cross-subject searches, so that; for instance, a keyword search on the word "energy" could bring together entries from social studies, science, and perhaps science fiction. Below is an example of note activity and groupings of information that successively incorporate more global concepts. The following notes were entered on CSILE's timeline during a unit on eyeglasses.

1001 BC-Magnifying Glasses Invented; 1268 AD-Eyeglasses Invented; 1500 AD-Glasses for Nearsightedness; 1780 AD-Bifocals Invented; 1888 AD-Contact Lenses Invented.

The explanation that the Arabs used marbles for magnifying glasses was attached to the "1001 BC note."

After reading these notes a student wrote "Glasses for nearsightedness invented in 1500. This came a long time after glasses for farsighted people. Why?"

While writing about "navigational instruments" a different student, working months later, noted that in 1608 a dutch eyeglass maker used the idea of holding two lenses a certain distance away and the telescope resulted from this idea. Galileo was, she noted, the first-known user of the telescope.

Over time, the database serves to cluster notes due to common themes (in this case "eyeglasses") and knowledge-structuring environments (in this case the timeline). Such clusterings lead to more powerful cross-domain understandings than might occur otherwise.

Many students use keyword searching when writing a summary note, so that they can see if others mentioned important information they might have missed. In future developments, it is intended that CSILE will incorporate intelligent database management that can display to students the interconnections among various notes and that can detect and call the students' attention to cross-subject relationships that the student might not otherwise be aware of.

Give Students More Responsibility for Contributing to Each Other's Learning

The emphasis here is on cooperative learning, which is not the same thing as cooperative task performance. It is easy to implement the latter through

computer activities, but it takes dedicated planning to achieve the former. For cooperative learning to occur, students must recognize what they are trying to learn, value it, and wish to share that value. It seems unlikely that computers can foster cooperative learning on their own, but they could play a role in a classroom culture where cooperative learning is encouraged. Educational software should help students recognize what it is they are learning and provide aggregate data that would allow them to monitor the learning progress of the class as a whole and not just their own progress.

CSILE application - Shared responsibility is perhaps the most important principle for achieving the overall objectives of computer-supported intentional learning environments. It has three aspects: 1) *direct contributions via the system*. Students respond to other students' ideas, requests for information, confusions, self-ratings, and so on. Teachers can base grades in part on the helpfulness of such responses, and (in future versions of CSILE) students will be able to rate how helpful they found other students' contributions. While some students rely on the privacy option, and search for only their own work, these strategies are gradually being given up, in favor of more open strategies. 2) *Preparation for class contributions*. Even in modern classrooms, where everyone is expected to contribute to the learning enterprise, it is usually only the teacher who gives prior thought to goals for a classroom session, to significant topics or questions for discussion, to problems or misunderstandings that need to be dealt with, and so on. Through CSILE activities, all the students can be involved in such preparatory thinking and this should significantly enhance the quality of classroom sessions. 3) *Acquiring higher-level executive control of learning processes*. By getting students to participate in activities that have been traditionally regarded as "teaching" rather than "learning" activities, CSILE can promote the higher levels of knowledge that are associated with being able to teach what one has learned [26].

CONCLUDING REMARKS

Students in grades five and six used CSILE daily for almost eight months. Students used the system to elaborate models and hypotheses, to delve into difficult texts, to seek deeper levels of explanation, to elaborate confusions, and generally to engage in processes thought to be beyond their years. Their veteran teachers thought their students would tire of the computer after the novelty wore off, but this clearly was not the case. CSILE remained a motivator for school work throughout the school term. CSILE is now in its second year of operation, and students from the first year return and report how much they miss it. For the teachers, CSILE is such an integral part of their instruction that they feel they would have to take a substantial step backwards if the computers were removed.

There is a great deal of development work needed before CSILE acts as an umbrella to other software to other software and incorporates prepackaged databases. It will be longer yet before it accomplishes its aim of fostering intentional learning across the school curricula and for children at all grade

levels. Nonetheless, the core design seems to be appropriate and flexible enough to accomplish long term goals. Ongoing research is aimed at continual testing of the system with children and adults. Not all users, of course, demonstrate the same sophistication in use. Continual refinements aim to take its most significant uses and make them commonplace.

ACKNOWLEDGMENTS

Many people have contributed to the work reported in this article. first we would like to thank the Ontario Ministry of Education for their generous support. They have provided the funds for software design, as well as supplying the two pilot classrooms with sixteen networked ICONS. The teachers who have collaborated with us, Chuck Laver and Jim Webb, have provided invaluable support and insights. They have, in turn, been supported by Ted Reinholdt, the Principle of Huron St. Public School and sixty-four patient and enthusiastic students from grade five and six. The project further has been made possible by a talented team of systems analysts who have built CSILE. This team includes Earl Fogel, Steve Hull, Alan Rosenthal, Peter Rowley, Orest Nahacziwec, and Russell Owen. We also wish to thank our researchers: Maria Yau, Alister Cumming, Anna Palamedes, Martin Ruck, Pam'la Ghent, Irene Rukavina, Clare Brett; graduate students: Evelyn Ng, Patricia Probert, Rob Higgins, Mary Bryson, Carol Chan; and support staff: Mary Ann Potter, Carolyn Calhoun, Denise King.

REFERENCES

1. E. R. Steinberg, Review of Student Control in Compute-Assisted Instruction, *Journal of Computer-Based Instruction*, 3, pp. 84-90, 1977.
2. C. Bereiter and M. Scardamalia, Intentional learning as a Goal of Instruction, in *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*, L. B. Resnick (ed.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, in press.
3. D. E. Lavin, *The Prediction of Academic Performance*, Wiley, New York, 1965.
4. S. S. Chipman, J. W. Segal, and R. Glaser (eds.), *Thinking and Learning Skills: Vol. 2. Research and Open Questions*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1985.
5. R. B. Davis and C. C. McNight, Modelling the Processes of Mathematical Thinking, *Journal of Children's Mathematical Behavior*, 2, pp. 91-113, 1979.
6. L. B. Resnick and R. Neches, Factors Affecting Individual Differences in Learning Ability, in *Advances in the Psychology of Human Intelligence*, 2, R. J. Sternberg (ed.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 275-323, 1984.
7. J. Larkin, J. McDermott, D. P. Simon, and H. A. Simon, Expert and Novice Performance in Solving Physics Problems, *Science*, 208, pp. 1335-1342, 1980.

8. A. L. Brown and J. D. Day, Macrorules for Summarizing Texts: The Development of Expertise, *Journal of Verbal Learning and Verbal Behavior*, 22, pp. 1-14, 1983.
9. J. J. Franks, N. J. Vye, P. M. Auble, K. J. Mezynski, G. A. Perfetto, J. D. Bransford, B. S. Stein, and J. Littlefield, Learning from Explicit vs. Implicit Text, *Journal of Experimental Psychology: General*, 111, pp. 414-422, 1982.
10. M. Scardamalia and C. Bereiter, Development of Strategies in Text Processing, in *Learning and Comprehension of Text*, H. Mandl, N., Stein, and T. Trabasso (eds.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 379-406, 1984.
11. -----, Fostering the Development of Self-Regulation in Children's Knowledge Processing, in *Thinking and Learning Skills: Vol. 2. Research and Open Questions*, S. F. Chipman, J. W. Segal, and R. Glaser (eds.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 563-577, 1985.
12. J. K Torgesen, The Learning Disabled Child as an Inactive Learner: Educational Implications, *Exceptional Education Quarterly*, 1, pp. 45-52, 1982.
13. A. L. Brown, J. D. Bransford, R. A. Ferrara, and J. C. Campione, Learning, Remembering, and Understanding, in *Handbook of Child Psychology: Vol. 3. Cognitive Development*, 4th Edition, J. H. Flavell and E. M. Markman (eds.), John Wiley and Sons, New York, pp. 77-166, 1983.
14. C. Bereiter and M. Bird, Use of Thinking Aloud in Identification and Teaching of Reading Comprehension Strategies, *Cognition and Instruction*, 2, pp. 131-156, 1985.
15. A. S. Paliscar and A. L. Brown, Reciprocal Teaching of Comprehension-Fostering and Monitoring Activities, *Cognition and Instruction*, 1, pp. 117-175, 1984.
16. M. Scardamalia, C. Bereiter, and R. Steinbach, Teachability of Reflective Processes in Written Composition, *Cognitive Science*, 8, pp. 173-190, 1984.
17. R. W. Lawler, Learning Environments: Now, Then, and Someday, in *Artificial Intelligence and Education: Vol. 2. Learning Environments and Tutoring Systems*, R. W. Lawler and M. Yazdani (eds.), Ablex Publishing, Norwood, New Jersey, pp. 1-25, 1987.
18. M. M. Riel, J. A. Levin, and B. Miller-Souviney, Learning with Interactive Media: Dynamic Support for Students and Teachers, in *Artificial Intelligence and Education: Vol. 2. Learning Environments and Tutoring Systems*, R. W. Lawler and M. Yazdani (eds.), Ablex Publishing, Norwood, New Jersey, pp. 117-134, 1987.
19. M. Scardamalia and C. Bereiter, Research on Written Composition, in *Handbook of Research on Teaching*, 3rd Edition, M. Wittrock (ed.), Macmillan, New York, pp. 778-803, 1987.
20. A. Collins, J. Brown, and S. Newman, Cognitive Apprenticeship: Teaching the Craft of Reading, Writing, and Mathematics, in *Knowing, Learning and Instruction: Essays in Honor of Robert Glaser*, L. B. Resnick (ed.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, in press.

21. A. L. Brown and A. S. Paliscar, *Guided, Cooperative Learning and Individual Knowledge Acquisition*, paper presented at the Conference on Cognition and Learning, Learning Research and Development Centre, University of Pittsburgh, 1985.
22. G. C. Wells, *Language Development in the Pre-School Years*, Cambridge University Press, Cambridge, 1984.
23. N. L. Stein and T. Trabasso, What's in a Story: An Approach to Comprehension and Instruction, in *Advances in Instructional Psychology*, R. Glaser (ed.), Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1982.
24. J. D. Novak and D. B. Gowin, *Learning How to Learn*, Cambridge University Press, Cambridge, 1984.
25. A. N. Whitehead, *The Aims of Education*, Macmillan, New York, 1929.
26. L. Shulman, Those Who Understand: Knowledge Growth in Teaching, *Educational Researcher*, 15, pp. 4-14, 1986.

Direct reprint requests to:

Dr. Marlene Scardamalia
Centre for Applied Cognitive Science
O.I.S.E.
252 Bloor St. W.
Toronto, Ontario, Canada M5S 1V6