DOCUMENT RESUME

ED 288 913 TM 870 739

AUTHOR	Scardamalia, Marlene; And Others
TITLE	Computer Supported Intentional Learning Environments.
INSTITUTION	Ontario Inst. for Studies in Education, Toronto.
PUB DATE	Apr 87
NOTE	10p.; Paper presented at the Annual Meeting of the American Educational Research Association (Washington, DC, April 20-24, 1987).
PUB TYPE	Speeches/Conference Papers (150) -- Reports - Descriptive (141)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	*Cognitive Objectives; Cognitive Processes; *Computer Assisted Instruction; *Computer Graphics; *Computer Software; Elementary Education; Feedback; Higher Education; *Intentional Learning; Learning Strategies; *Program Implementation; Student Evaluation; Student Participation

ABSTRACT

Computer Supported Intentional Learning Environments (CSILE) is a project using computer software to promote the constructive processes involved in intentional learning. CSILE enables groups of students to build a collective database (knowledge base) of their thoughts, in the form of pictures (created by students using a color graphics editor) and written notes. CSILE was initially developed for university and graduate-level students; current research focuses on 64 students in two sixth-grade classes. The report offers 11 principles applicable to a variety of educational software and then discusses the application of that principle to the project: (1) make knowledge-construction activities overt; (2) maintain attention to cognitive goals; (3) treat knowledge lacks in a positive way; (4) provide process-relevant feedback; (5) encourage learning strategies other than rehearsal; (6) encourage multiple passes through information; (7) support varied ways for students to organize their knowledge; (8) encourage maximum use and examination of existing knowledge; (9) provide opportunities for reflectivity and individual learning styles; (10) facilitate transfer of knowledge across contexts; and (11) give students more responsibility for contributing to each other's learning. CSILE examples cover both how principles were being realized, and how novices can at times circumvent the goal of eliciting extra constructive effort. (LPG)

COMPUTER SUPPORTED INTENTIONAL LEARNING ENVIRONMENTS

Marlene Scardamalia, Carl Bereiter, Robert S. McLean,
Jonathan Swallow, Earl Woodruff

In: A. Collins (Chair), Strategies for teaching thinking skills with interactive technologies. Symposium conducted at the meeting of the American Educational Research Association, Washington, 1987.

(1987). In A. Collins (Chair), Strategies for teaching thinking skills with interactive technologies. Symposium conducted at the meeting of the American Educational Research Association, Washington.

Computer Supported Intentional Learning Environments

Marlene Scardamalia, Carl Bereiter, Rob rt S. McLean,
Jonathan Swallow, Earl Woodruff

Background

Intentional learning requires constructive effort consisting of goal-setting, identifying and solving problems of understanding, connecting old and new knowledge, constructing and testing inferences, and monitoring and evaluating learning. 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 higher order thinking activities, 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.

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. Current research focuses on two grade 6 classes. There are a total of 64 students who have been using CSILE three or more times each week, since November. CSILE is being made available to them on 16 networked ICON microcomputers, 8 in each classroom.

Data have been collected through case study observations of the students, and thesis research, in addition to the notes and pictures passively collected by CSILE itself. These data include many demonstrations of high constructive effort expended by the students.

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 higher-order learning and thinking abilities. One reason such specifications are neede 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 desirable tc 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. 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, as well as examples of novice strategies which have allowed some students to resist expending extra constructive effort.

Principles

1. 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 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. In some cases patterns can be seen where students begin with some new learning, generate questions and plans for the investigation of those questions. However, the novice strategy users in many cases are either choosing inappropriate thinking type labels, or just choosing them at random to fulfill the requirements for sending a note to the database.

2. 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 recent study by R. Lloyd (OISE course project) compared an expert and a 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 by student-designed icons. It will be supported by prompts to indicate cognitive goals, plans for pursuing them, and target dates. 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.

3. <u>Treat knowledge lacks in a positive way</u>. 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 about what they are studying is one of the main things students do in CSILE notes. Class activities and 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. 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 cognitive developmental levels in question-asking and ways to promote progress toward higher levels that can be implemented on CSILE.

4. <u>Provide process-relevant feedback</u>. 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 future versions, however, students will be able to obtain reports directly. In addition, there will be provision for a variety 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. 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 seem to be sensitive to the number of attempts made to generate useful ideas, as well as the resources available, when rating their partners' efforts. A finished product is not necessary for high ratings. However, in some cases students seem to be acknowledging form rather than substance. Modelling of more expert thinking operations seems needed to provide a better basis for comparison.

5. 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 (Brown et al., 1983). Drill-and-practice approaches in educational software naturally 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. 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.

6. 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. The same note may appear at various times through database searches using different search criteria, may appear on a timeline, and may appear on one or more charts. Scratchpad, note editing, and copying functions encourage students to reprocess previously recorded information. In some cases the students seem content to indicate what they found interesting, or which of the presented facts they did or didn't already know. However, 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.

7. 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 any organized fashion to present it to them and take it from them in that format; and 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 8 or beyond categorization is already their natural way of organizing knowledge and what they need most is help in growing beyond that very 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 (Scardamalia & Bereiter, 1984). There are many alternative ways of structuring knowledge, such as time-lines, graphs, maps, narrative sequences, story grammar structures (Stein & Trabasso, 1982), concept nets (Novak & Gowin, 1984), 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 time-line to which students can attach notes. Students will utilize all of these presentation methods for a unit, such as frog dissection. However, unless they are given explicit instructions, some students tend to produce unrelated pictures, or a single, long written note about a topic. Future developments will include other types of knowledge organizations mentioned in the preceding paragraph.

8. 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, labeling 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. Some students have difficulty finding value in other peoples notes. One students reported that "they all just said the same thing, just differently." Many students do work from others' notes, but rather than copying, they often feel a responsibility to say something new.

9. 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 of 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 a 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 In the worst case, CSILE provides an opportunity to waste time. However, the goals of this design principle have been realized through students' use of the scratch pad for making rough notes, the use of the facility for placing notes on hold while other ideas are pursued, and the use of a feature which allows notes already in the database to be edited.

10. <u>Facilitate transfer of knowledge across contexts</u>. The compartmentalization of learning into school subject areas has long been blamed for students' failures to transfer knowledge to new contexts (Whitehead, 1929), 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. CSILE search and retrieve facilities allow students to get only their own notes back from the database. Some students rely on this strategy when working from existing information, and to some extent missing the connections between information in the system. However, many students do use keyword searching, for example, when writing a summary note, so 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 student's attention to cross-subject relationships that the student might not otherwise be aware of.

11. <u>Give students more responsibility for contributing to each other's learning</u>. 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 that they are learning and what it is they are learning (see points 1 and 2), and in addition it could provide aggregate data that would allow students 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: (a) 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. (b) 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. (c) 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 (Shulman, 1986).

Closing

Technologies similar to CSILE are being developed at Xerox (Note Cards), Brown University (Hypermedia), and Carnegie Mellon University (Andrew/Tutor). CSILE does not make use of the advanced technologies supporting these other systems, however, it does have several advantages. Because CSILE employs more readily available technologies, a working system was created in a relatively short time, and has already undergone extensive field testing. Networking, and CSILE's emphasis on cooperative group learning gives it the potential to become a much richer system than those which provide more support for individual work. We feel that it is this cooperative environment which has contributed most heavily to its success. Many students have been able to adapt their novice strategies to allow themselves to be content to use CSILE in more superficial ways. However, ongoing research is aimed at making these strategies less effective, and helping students acquire more expert strategies for learning.

References

Brown, A. L., Bransford, J. D., Ferrara, R. A., & Campione, J. C (1983). Learning, remembering, and understanding. In J. H. Flavell & E. M. Markman (Eds.), Handbook of child psychology: Vol. 3. Cognitive development (4th ed.). (pp. 77-166). New York: John Wiley & Sons.

Novak, J. D., & Gowin, D. B. (1984). Learning how to learn. Cambridge: Cambridge University Press.

Scardamalia, M., & Bereiter, C. (1984). Development of strategies in text processing. In H. Mandl, N. Stein, & T. Trabasso (Eds.), Learning and comprehension of text. (pp. 379-406). Hillsdale, NJ: Lawrence Erlbaum Associates.

Shulman, L. (1986). Those who understand: Knowledge growth in teaching. Educational Researcher, 15, 4-14.

Stein, N. L., & Trabasso, T. (1982). What's in a story: An approach to comprehen- on and instruction. In R. Glaser (Ed.), Advances in instructional psychology. (pp. 213-267). Hillsdale, NJ· Lawrence Erlbaum Associates.

Whitehead, A. N. (1929). The aims of education. New York: Macmillan.