5 From Cognitive Modeling to the Design of Pedagogical Tools

Kurt Reusser
University of Zurich

DESIGNING TECHNOLOGY-BASED TOOLS FOR LEARNING AND INSTRUCTION: A DIDACTIC TASK

Since Comenius' (1659) *Orbis Pictus*, educators have debated how learning and teaching can be improved and facilitated by the design of instructional media, methods, and materials. For the last several decades, this debate has also encompassed the use of electronic media and technology. Research on computer-aided instruction goes back to the end of the 1950s, whereas the educational use of computers began in the 1970s. Inspired by Skinner's behaviorist conception of learning, which called for the successive approximation of carefully decomposed, sequentially presented, and reinforced tasks, a broad variety of instructional programs for computers and textbooks was developed. The pedagogical idea behind this early phase of computer-assisted, or programmed, instruction was to provide for an improved efficiency and individualization of learning. Each student should be able to acquire skills and domain knowledge at his or her own pace by practicing skills according to his or her abilities. However, because of both the minimal power of the hardware and the underlying theory of learning, which failed to foster conceptual learning and comprehension, most of the early programs (in fact, little more than electronic page-turning devices) did not meet the expectations of educators and the requirements of any demanding type of instruction.

Three decades after the cognitive revolution in psychology, and with the advent of powerful and inexpensive computers, computer-assisted instruction—or the more ambitious intelligent tutoring system (ITS)—has again become a central topic of research in the field of education. Behind the present efforts to make learning more efficient, motivating, and individually adaptive, there are widespread and, in part, romantically high expectations about the role and didactic power of computers in education and instruction. Computers seem to be conceived of as not merely tools, among other tools, for future teaching and learning, but, for example, as (a) intelligent and mind-empowering prosthetic devices, (b) amplifiers of human intelligence (allowing one to do more things faster and more accurately, (c) reorganizers of mental functioning (leading to a shift in the kinds of problem types and levels of difficulty that can be managed and in the types of methods used to solve problems), or (d) even instruments of cultural redefinition (Table 5.1). Computers are perceived as personal electronic teachers—as intelligent teaching systems associated with the vision of the computer as the "future dominant delivery system in education for almost all age levels and in most subject areas" (Bork, 1985, p. 1).

Because computer-supported instruction is a rather expensive way to teach, one should consider the purpose for which it will be used in education. Because of the elevated costs of building high-quality software, one has to think carefully about what things are worth doing with computers as well as what can be done best with them.

Thus, the design of any computer-based educational system should be based on two foundations: on content-specific research on learning and comprehension, and on a pedagogical model of the learner and the learning process. The latter must include an account of how computers are to be integrated into the classroom, as well as the use of technology-based pedagogical tools in general.

TABLE 5.1
Metaphors and Views Concerning the Role of Computers in Learning and Teaching

Computer Uses
• systems for automating education (Anderson, Boyle, Corbett, & Lewis, 1990)
• idea amplifiers; personal electronic teachers (Brown, 1984)
• cultural amplifiers; part of cultural tool kit; prosthetic devices (Bruner, 1986)
• reorganizers of mental functions; extracortical organizers of thought; a medium that helps transcend the limitations of the mind (Pea, 1985, 1987)
• instruments of cultural refefinition (Bruner, 1986; Pea, 1987)
• intelligent tutoring systems (Anderson, Boyle, Farrell, & Reisser, 1984; Ohlsson, 1986; Sleeman & Brown, 1982)
• intelligent learning environments (Brown, 1984)
• idea amplifiers; facilitators of learning (Brown, 1984)
• amplifiers of the mind (Bruner, 1986; Pea, 1987)
• mirrors of the mind (Brown, 1984)
• an electronic workbook; a tool for learning through reflection (Collins & Brown, 1988)
• a medium for experiential learning (Lepper & Gurtner, 1989; Papert. 1980)
• a foundation for new learning cultures (diSessa, 1989)
• cognitive tools (Lajoie & Derry, 1993; Pea, 1987)
• partners in cognition (Salomon, Perkins, & Globerson, 1991)

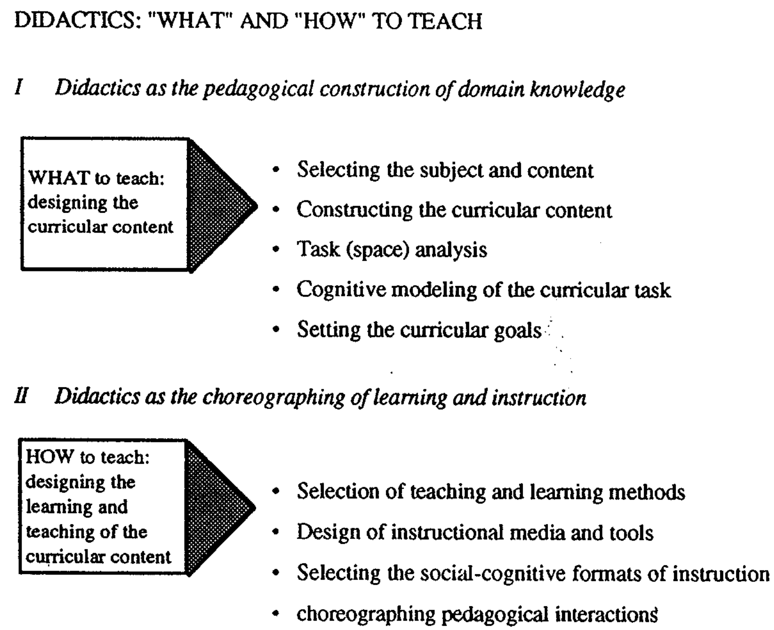


FIG. 5.1. The dual *bildung* theoretical task of didactics as pedagogical knowledge construction and choreography of learning and instruction.

In contrast to any type of opportunistic, technology-driven design, and in accordance with the concept of *didactics* as a design science (Simon, 1981; Wittmann, 1992), the design of computer-based teaching and learning environments should be considered a genuine didactic task. In the original sense of the classical theory of *Bildung* (Klafki, 1963; Weniger, 1952; Willmann, 1889/1957), the concept of *didactics* (Fig. 5.1) includes the questions of what to teach and how to teach. The former is concerned with selecting, analyzing, and modeling the curricular content and goal structure. The latter relates to decisions about pedagogical methods, presentation forms, and media use, as well as an understanding of patterns of sociocognitive interactions. It must be based on a philosophical and developmental view of the learner and the nature of learning processes.

Thus, at the heart of any pedagogical design is the didactic analysis (Klafki, 1958) of key concepts, structures, and representations; learning methods, skills, and strategies; as well as patterns of instruction related to a task or domain. As is often observed, insufficient attention to the sound cognitive-didactic decomposition of a curricular task can ruin costly and well-meant instructional efforts.

A THEORY OF MATHEMATICAL STORY-PROBLEM COMPREHENSION AND SOME INSTRUCTIONAL IMPLICATIONS

Mathematical text, word, or story problems (MSPs), which are used to assess students' application of mathematical knowledge and skills, consist of two interwoven semiotic worlds: a storylike description of a nonmathematical situation or event, and an implicit web of mathematical relations. Thus, cognitive-didactic analysis with respect to solving MSPs has to focus on the interplay among language processes, the understanding of a denoted situation, and mathematical problem solving, as well as on students' comprehension strategies, errors, and difficulties. Because any comprehension of an MSP is mediated by the way the problem is presented linguistically, and with respect to the situational context in which a mathematical structure is embedded, didactic task analysis must focus on the role of problem language and real-world knowledge (i.e., on the broad variety of presentational factors involved in problem posing and wording; Staub & Reusser, 1995).

Situation Problem Solver

Our theory of mathematical word- or story-problem comprehension (Reusser, 1985, 1990) integrates work of Piaget (1947) and Aebli (1980–1981) on the nature and development of mathematical thinking, of van Dijk and Kintsch (1983) on discourse processing, and of Kintsch and Greeno (1985) and Cummins, Kintsch, Reusser, and Weimer (1988) on arithmetic word problems. We have developed a production rule model that is based on a decompositional analysis of the tacit knowledge, the difficulties of the language and situation comprehension processes, and the skills involved in solving word problems. The computational model takes elementary word problems as natural language input (within some range of complexity) and simulates the processes of their understanding and solving by accessing linguistic and general world knowledge and applying various kinds of strategies. Tracing the comprehension, mathematization, and solution process as leading from text to situation to equation, several mutually constraining levels of mental representation (textual, situational, and mathematical) are generated by the situation problem solver (SPS; see Fig. 5.2):

1. a text base as a propositional representation of the task extracted from the textual input;
2. an episodic situation model as a goal- and task-specific, qualitative representation of the elaborated nonmathematical (problem) situation or event denoted by the story text;
3. a mathematical problem model capturing the inferred gist of the mathematical situation, that is, the elements and relations of the episodic situa-

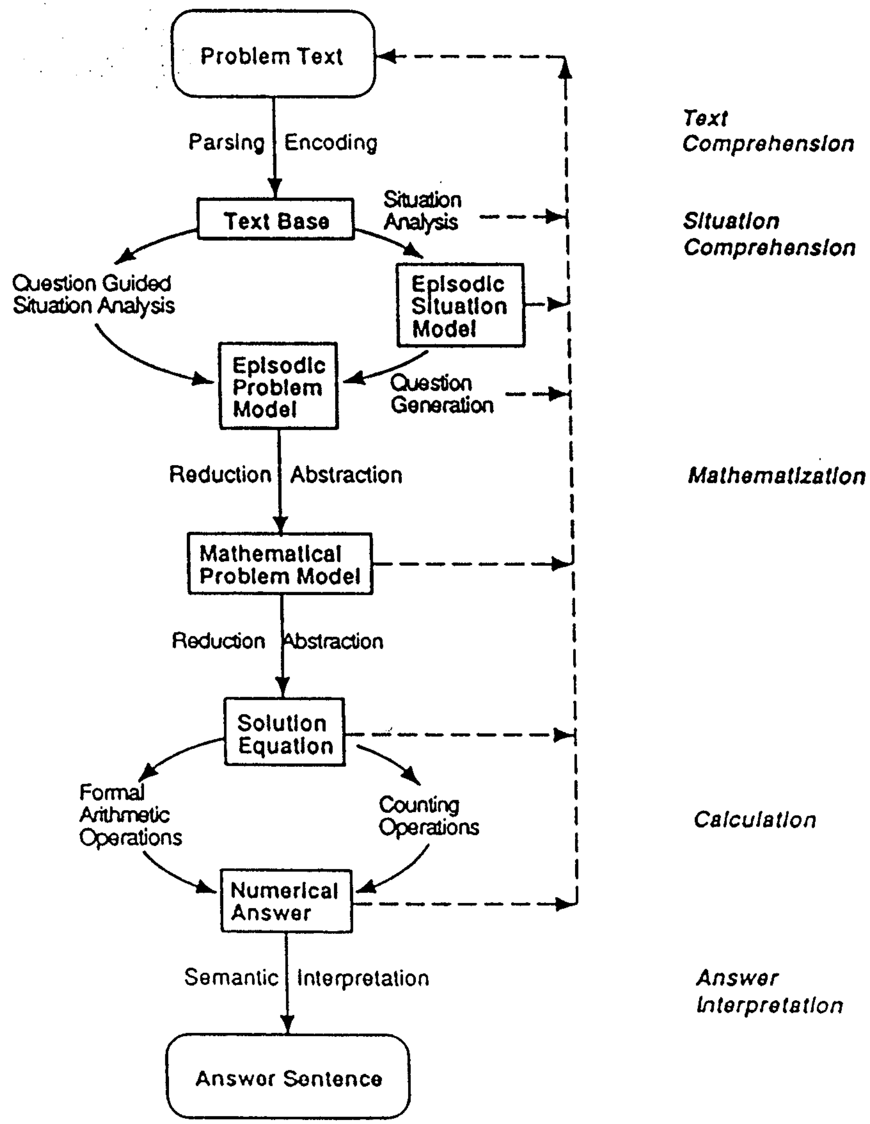


FIG. 5.2. Types and levels of mental representations generated in the process model SPS during problem comprehension and mathematization.

tion model that are relevant from the point of view of a mathematical question to be answered; and

4. a formal equation as the densest form of the mathematical situation inherent in a word problem.

According to our process model of problem comprehension (for more details, see Reusser, 1989a, 1990), the crucial steps in successful problem comprehension are those that lead to the formation of an adequate qualitative and—derived from it—a semiquantitative situation model, or problem model. Successful comprehension processes consist of a sequence of inferential and elaborative steps that are elicited or induced by the wording of the problem and guided by an explicit problem question. The semiquantitative situation model in SPS is seen as a necessary bridge in meaningful problem comprehension, connecting to both the initial linguistic input (problem text) and the abstract expression of the inherent mathematical relations (symbolic equation formulae).

Faulty problem solving can be simulated in SPS by introducing deficient knowledge into the program, which leads to erroneous problem representations. Empirical evidence gathered so far (Reusser, 1989b; Staub & Reusser, 1995) supports the view of competent comprehension and solving of MSPs as rooted in the development of language comprehension strategies, rather than in the development of mathematical skills alone. Presentational factors (Staub & Reusser, 1995), that is, the linguistic and pragmatic forms employed by the authors of MSPs, have been repeatedly identified as a critical and diverse source of problem difficulty (cf. Cummins et al., 1988; De Corte, Verschaffel, & de Win, 1985; Stern, 1991; Stern & Lehrndorfer, 1992).

EDUCATIONAL IMPLICATIONS

What this implies for instruction with SPS is that mathematical word problems should be taught as problems of comprehension. This means that MSPs should be treated as a significant object of discourse and reasoning in mathematics education. With the goal of achieving basic mathematical literacy, it is not enough (but often the case in mathematics education) to teach students formal structures and isolated procedural skills. To do so is to develop decontexualized symbolic structures and proceduralized operational skills, both types of knowledge disassociated from deeper comprehension and situational referents and application. Following this line of argument, it is not enough to teach students to solve MSPs, for instance, by means of keyword-driven and structurally blind "direct-translation" rules (cf. the STUDENT program of Bobrow, 1964; Paige & Simon, 1966), that is, picking up and mapping syntactic cues into algebraic language (cf. Nesher & Teubal, 1975). In contrast, in instructing applied formats of mathematical problem solving (e.g., word problems), mathematics education should

focus on the bridging of abstract, symbolic problem solving with concrete, situational reasoning. That is, formal, symbolic expressions that result from word problems should become transparent and situationally meaningful by rooting them in their corresponding events—in the real-world situations from which they are derived.

For the process of teaching, this requires formats and notations by which mathematical word problems can be overtly represented, inspected, reflected upon, and communicated in appropriate ways. Students need instruction on how to use representational systems and languages (and how to generate their own formats and notations) to express their situational and mathematical understanding of a problem in meaningful and cognitively efficient, often semiquantitative ways. Because the use of representational notations belongs to the tool kit of expertise in any domain, powerful (especially culturally shared and canonical) general notations and domain-specific representations—together with representation-building skills—need to be taught and conveyed to students as a significant part of achieving mathematical and scientific literacy (Mitman, Mergendoller, Marchman, & Packer, 1987).

Computers have the potential to serve as powerful representational tools (Reusser, 1993). Before turning to the description of a tutoring environment that was derived from our cognitive modeling work, our instructional design philosophy, including a set of principles, is outlined. Although these principles are consistent with the described cognitive simulation work, they are—as is discussed—only in part strictly derived from it.

FROM COGNITIVE MODELING TO THE DESIGN OF A TECHNOLOGY-BASED EPISTEMIC TOOL

The Romantic Dream of "Automating Education"

What are the pedagogical ideas associated with the high hopes—the "computer dream" (Lepper & Gurtner, 1989)—that accompany the new generation of instructional technology? I think they have multiple roots, with an emphasis on three topics. First, there is the perennial effort for more individualization and adaptivity in learning and instruction (an idea than can be traced back far in the history of education). A second motive is learning by doing, or learning from experience (an idea going back beyond Dewey and Piaget, to Rousseau and Pestalozzi): Computers should provide students with supportive contexts for discovery and experiential learning. A third idea that has been associated with educational technology, especially with environments such as *Logo* (Papert, 1980) or *Boxer* (diSessa, 1989), is an emphasis on cognitive skills, or the expectation that properly designed systems will contribute to the development of higher order learning and thinking skills. This issue relates to the 19th-century debate on *formale* versus *materiale Bildung* (formal vs. content-oriented education) in the German-speaking history of schooling (Lehmensick, 1926), and to the problem of formal discipline and training of mental faculties in Anglo-American tradition (Mann, 1979).

An ambitious type of computational system—with which high expectations about didactic power and value have been associated—is the intelligent tutoring system (ITS; Anderson, 1989; Mandl & Lesgold, 1988; Ohlsson, 1986; Sleeman & Brown, 1982). Such a system assumes that computers might eventually function as adaptive and self-sufficient substitutes for intelligent human teachers. The architecture of an ITS consists of four components (see Table 5.2): an expert or knowledge component (containing the model of a specific content or domain), a learner model associated with diagnostic capabilities (allowing inferences about what a student thinks and knows), a tutorial planning and teaching module, and an easily manipulable user interface. The first three components refer to the presumed threefold intelligence of an ITS, namely, task representation (modeling the content), student diagnostic (learner modeling), and didactic (teaching) intelligence. Obviously, it is not a trivial task to build a system whose goal is to provide learners with individually tailored feedback and adaptive instructional supports on the basis of a constantly retuned student model. Such a system

TABLE 5.2
Four Modules of an ITS Architecture

Module	*Description*
Expert domain-knowledge module	Explicit model of the domain • representation of the domain knowledge of experts:facts, concepts, strategies, skills, tasks • expert program that solves problems in the domain
Learner module	Cognitive simulation of the learner • fine-grained cognitive diagnosis and evaluation of the student's understanding: knowledge states and reasoning processes, bugs, misconceptions • Identification and dynamic representation of the emerging knowledge and skill of the student
Tutoring module	Orchestration of didactic interventions and control • modeling the teacher-student interaction • adaptive selection and presentation of task material • adaptive diagnostic feedback and tutorial help • tutoring activities: explanation, advice, remediation, testing
User-interface module	Ergonomy of interaction • communication/interaction between user and system • direct manipulation, graphics, natural language

implies no less than putting a fully competent (ideal) teacher inside a computer. Despite some notable success stories (cf. Anderson et al., 1990; Shute & Glaser, 1990), there are reasons to be skeptical about both the feasibility and the wisdom of intelligent systems based on full system control and deep student modeling.

Feasibility. Based on cognitive modeling (including our own on mathematical word problems), and by taking into account the state of the art in knowledge engineering and cognitive psychology, we conclude that machine tutoring based on cognitive simulation of the student is still not possible across a full range of tasks and in open-ended domains. Fuzzy language and the enormous amount of qualitative world knowledge and situated-reasoning skills that are required form an insurmountable obstacle. Our psychological theories about learning, knowing, and comprehension are still inadequate to enable computer modeling of qualitatively rich and contextualized knowledge processes.

Should educators worry about this situation? Not too much, I think. First, evidence shows that human teaching is not based on fine-grained diagnostic behavior as much as the ITS philosophy presumes (i.e., expert teachers do not carry out extensive cognitive diagnosis; cf. McArthur, Stasz, & Zmuidzinas, 1990). Second, even if the ITS is not within the reach of today's knowledge technologies,[1] instructional systems—where a machine sensitively adapts its teaching on the basis of a fine-grained learner model—might still be seen as a long-term goal (Kintsch, 1989). Third, there are alternative and more robust ways to support and facilitate learning and problem solving through interaction with a computer.

Pedagogical Wisdom. Furthermore, one can question a didactic approach to learning—with or without computers—that takes students by the neck and forces them down some predefined, presumably efficient solution path,[2] without allowing students to make errors or to get lost, and without encouraging planning, goal setting, diagnosis, and self-assessment. This skeptical view is consistent with remarks made by Scardamalia, Bereiter, McLean, Swallow, and Woodruff (1989), who called on the active and intelligent learner, not the computer system—which is merely seen as a facilitating tool—to perform the diagnosing, goal setting, and planning (see also De Corte, chap. 7, this volume).

Alternative and less directive (un)intelligent computer-based systems for learning and instruction can contribute to the ultimate goal of developing virtually autonomous learners and reflective problem solvers.[3] This does not require

[1]At best, today's cognitive modeling techniques provide educational researchers with a heuristic tool for further exploring the rich phenomenology of cognition and instruction (cf. Reusser, 1990).

[2]This seems to be the underlying pedagogy reflected by some principles of the ITS (cf. Anderson et al., 1984).

[3]For a discussion of unintelligent tutoring, compare Nathan, Kintsch, and Young (1992).

intelligent systems, but rather facilitative and flexible structures, that is, didactic supports that, instead of being located solely in the computer, reside in many brains and machines (Dillenbourg, chap. 9, this volume). The burden of providing intelligence should not be borne by a computer system. Instead, intelligence is distributed across the entire didactic setting and consists of the combined pedagogical and domain-related intelligence of teachers, learners, contexts, and technology-based tools.

The Myth of the Learner as a Radical Constructivist

Underlying the ideas of individualization, cognitive skill enhancement, and didactic adaptivity, which are driving current research on the development of educational technology, there is an even deeper motive or presupposition that is having an impact on the design of learning environments: the fuzzy issues of *constructivism*—often called *radical constructivism*—and *learner autonomy*, which are currently in vogue. Both concepts are infiltrating the psychopedagogical zeitgeist. Ultimately philosophical and epistemological in nature, these concepts have a long history in philosophy and education. Epistemological constructivism—referring back to the work of Piaget (1950) and his philosophical predecessor, Kant (1781/1965), who introduced the doctrine of constructivism—seems to have found its pedagogical complement in a set of instructional methods that appear under labels such as *inductive, experiential, self-directed,* or *discovery learning*. Pedagogical or didactic constructivism relates to the perennial history of progressive education since Rousseau's (1762/1951) *Emile*, although the relation seems to be conceptually vague and historically complex.

According to the doctrine of epistemological constructivism that is implied, for example, in Piaget's cognitive–developmental psychology, the ideal learner, including the child, is seen as a spontaneously active, volitional learner, as a discoverer and explorer, and as a virtually autonomous subject who shapes his or her mind and conception of the world as the result of intentional socio-cognitive activity.

Epistemological constructivism—considered from a cognitive instructional point of view—rests on the fundamental assumption that no learner can be forced by any teacher, tool, or material to an insight, to an understanding of a concept, or to become intrinsically involved in deeper learning. That is, the child's learning is always fundamentally constructive in the sense of his or her basic epistemic activity. According to Piaget, any developmental change, as well as any knowledge construction, is a matter of building relationships (*mettre en relation*), thus a genuine mental structure-building activity that every child always has to do on his or her own, and that nobody ever can do for the child.

Many educational researchers, however, go beyond the epistemological constructivism that is broadly accepted today. As proponents of what is labeled *radical constructivism,* they adopt a romantic and almost mythical view of the

self-constructive nature of children's minds, as well as an idyllic and over-idealized view of the societal task of learning and enculturation in a mass schooling system. Opposing all forms of directly guiding instruction, radical constructivism, in some of its proposed didactic consequences, comes close to *radical progressive education,* which is historically documented by more dead ends and failures than success stories (cf. Oelkers, 1989).

Assuming that all children possess almost unlimited capabilities, and viewing the knowledge and skills they acquire as emergent properties of largely nondirective interactions between children and more knowledgeable others, radical constructivism—if mapped into the didactic practices of nondirected instruction—may run the risk of underestimating the essential role of direct and indirect guidance of good teachers and carefully designed cognitive tools.

The following remarks are based on less optimistic assumptions regarding the constructive nature of learning with regard to the societal task of enculturation. First, adopting a constructivist view of the psychopedagogical nature of learning and comprehension must not lead us to equate the process of schooling with inductive discovery learning or exploratory activities under the almost unrestricted control of the learner. Children will not construct or discover in a few hours what has taken our culture years or centuries to develop. Discovery learning, in any demanding sense, is very slow; it mainly explains how cultures develop and change. School learning in a mass society, however, must be accelerated, with the goal of continuing and preserving culture and cultural identity across the time course of generations.

At the same time, we would not wish to underestimate the importance of other constructivist assumptions. These include a high regard for independent learning and the essential role of teachers in this process: as adaptive structural and procedural role models, as domain experts and scaffolds for expert learning, and, more generally, as impulse givers and facilitators of learning. Thus, to provide various forms of direct and indirect structural and procedural assistance is at the heart of professional teaching. This activity is equivalent to the pedagogical design of learning environments, and is in accordance with the concept of didactics as a design science (Simon, 1981; Wittmann, 1992).

Cognitive Tools for Intentional Learners: Six Design Principles

In this section, six principles for didactically intelligent computer-based tools are briefly sketched (for further elaboration, see Reusser, 1993). The principles help make up for the lack of attention received by two issues in the past: didactic learning theory and cognitive-instructional task analysis of content.

1. Design-intelligent technologies as cognitive tools for thoughtful teachers and learners. In contrast to technology-driven systems, cognitive tools should be used as means to pedagogical (didactic) ends or goals.

2. Stimulate and facilitate students' efforts toward domain-knowledge construction, understanding, and skill acquisition by providing expert procedural and domain conceptual (structural) assistance. This means implementing the qualities of expert learners and domain-knowledge experts—the latter representing the structure of a discipline—as instructional tools. Modern computers—with their direct-manipulation graphics interfaces—are ideally suited to provide representational and procedural support for students.

3. Provide students with intelligible representational tools of thought and communication. To understand a concept or a domain structure is to represent and express it in appropriate ways. As Simon (1981) remarked, reviving the spirit of Wertheimer (1945): "Solving a problem simply means representing it so as to make the solution transparent" (p. 153). Efficient and cognitively plausible representation of content is a fundamental problem for every theory of knowing and instruction. Effective representational skills, formats, and notations that constrain and support qualitative, semiquantitative, and quantitative reasoning in all scientific disciplines are part of the indispensible tool kit required for thinking and problem solving. Therefore, to teach culturally shared representational forms, and generic and domain-specific representation-building skills, is a significant and critical goal of instruction.

4. Provide as much learner control as possible and as much control of the learner as needed—or provide learners with some guidance according to the principle of *variable control and minimal help* (Aebli, 1961/1987). In contrast to the principle of *immediate feedback* in Anderson's (1989) teacher-centered approach to intelligent tutoring[4], the minimal-help principle states that learner control should be optimized by letting the system intervene only if help is needed and/or requested by the student. By making a virtue of necessity, the principle also adapts to rather limited capacity of most systems for student diagnosing and modeling.

5. By their potential as extracortical mirrors of the mind (Pea, 1985), computers should allow students to express and communicate their mental models and to reflect on their own processes and products of learning. Computers have the unique potential to provide learners with a powerful medium for representing, visualizing, and reflecting on their own mental models, as well as their learning and thought processes, including those of learning partners in collaborative learning. The long-term use of computers as tools for reflection on one's learning in different domains and disciplines may eventually lead to the overall reflectivity that characterizes mature learners and problem solvers.

6. Extend computer-based instruction from individual to cooperative contexts of learning. A questionable feature of our schooling culture is that students are

[4]"An error comes close to being a necessary and sufficient condition for tutorial intervention" (Anderson, 1989, p. 343).

treated almost exclusively as solo leaners (Bruner, 1986). However, intelligence is not a property of the mind alone, but rather a quality distributed among all components—agents (brains), tools (machines), and practices—of a learning environment (see also Dillenbourg, chap. 9, this volume). Therefore, because it is unlikely that computers will become truly adaptive cognitive partners or sensitive coaches in the near future, they should be integrated into the didactic settings of classrooms.

HERON: A COGNITIVE TOOL FOR UNDERSTANDING AND SOLVING MATHEMATICAL STORY PROBLEMS

We developed a cognitive tool, called HERON (after the Greek mathematician who was among the early inventors of mathematical word problems), based on our cognitive modeling work and on the previous cognitive-instructional principles. HERON is a mouse-driven, graphics-based problem-solving tool for understanding and solving complex mathematical story problems. HERON helps students from Grades 3 through 9 identify, conceptualize, and express the relevant pieces of information in a problem, and supports the reified planning and construction of a mathematical problem model, including the derivation of an equation.

Solution Trees

The design of conceptually faithful and cognitively efficient representations as instruments of thought and communication (Kaput, 1989) is more than a prerequisite pedagogical task, and far more than just ad hoc and tricky didactic art work. The issue of domain-representational form(at)s and notations touches the fundamental epistemological and ontological (cf. Greeno, 1983) questions of what constitutes our psychological building blocks and forms of knowing and thinking (Aebli, 1980–1981), that is, our most effective and satisfying cognitive models and activities.

HERON uses a graphical format for problem representation and planning called *solution trees*—a conceptual tool developed by Aebli, Ruthemann, and Staub (1986) for use with paper and pencil and employed in their empirical work. A similar format was developed by Derry and Hawkes (1989).

Solution trees provide students with a constraining format for planning and generating their solutions and, at the same time, for expressing their understanding of the problem in situational and mathematical forms (cf. the previous design principles). They consist of dynamically linked entities that capture the dual situational and mathematical deep structure implied in a broad range of story problems. Solution trees function as flexible, transparent, and visually inspectable tools for semiquantitative reasoning that allow the simultaneous grasping of both the semantic and the mathematical deep structure denoted by a problem text. Thus, solution trees are a type of the cognitively efficient notational formats proposed in our framework for cognitive instructional representation (Reusser, 1993).

Solution trees allow the problem solver to view together the semantic and underlying mathematical structure denoted by a problem text. Their value from an instructional point of view is that they invite and force students to focus on and visualize the logicomathematical deep structure of a problem, using a constraining format that relates problem comprehension to both the surface structure of the episodic problem text and its implied quantitative structure.

Working With HERON: An Example

Currently, HERON tutors any word or story problem that can be expressed by a solution tree. There are two implementations of the HERON environment: one is written in Loops (Kämpfer, 1991) and runs on a Xerox 1186, and the other is used in classrooms and runs on DOS machines (Stüssi, 1991).

HERON is designed so that it takes third graders about 20 minutes to become familiar with a mouse-driven interface, giving students a great deal of control over how to work with the system. To demonstrate its functioning, we use an example.

The specific problem that we assume a student decides to solve is depicted in the upper right window in Fig. 5.3. After carefully reading the problem text, the student is asked by the system to set up and instantiate problem-specific situation units or elements represented visually as boxes with the three value entries: *situation concept, quantity,* and *dimension* (unit of measurement). The generation of the boxes for quantities given in the problem text is done by highlighting a number or number placeholder in the text with the mouse cursor.

When quantity information (e.g., *15*) is selected, the system creates a box with the selected number filled in. The student then fills in a unit of measurement (e.g., *days*) and a label for an adequate situation concept (e.g., *remaining time until birthday*) that interprets the quantity in terms of a qualitative situation model. Both the dimension and the situation-concept label are selected from menus that can be activated by the mouse (cf. the problem-specific, situation-concept menu in Fig. 5.3). After the box representing a situation element has been completed, the student can select and interpret a new piece of quantitative information.

After instantiating two or more situation elements, the student can generate the first relational element of his or her solution tree (Fig. 5.3 and 5.4). To instantiate a first triadic-relational schema, for example, with the purpose of computing the number of rows still to be knitted, the situation units (boxes) labeled *length of the scarf* and *number of rows knitted so far* are selected, moved to the upper left corner of the screen, and linked together by a menu-selected

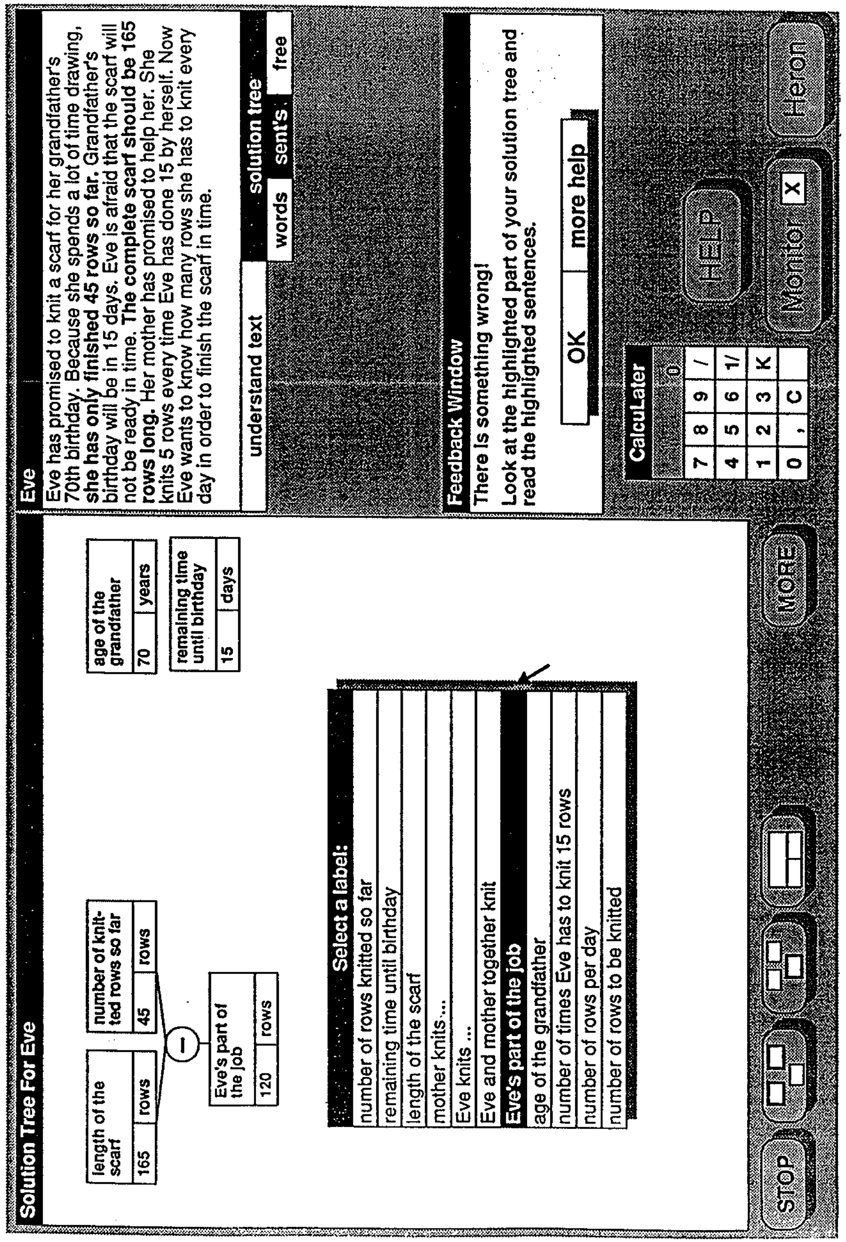


FIG. 5.3. Selecting and interpreting relevant quantitative information; beginning tree construction; current step: selecting an interpretive situational concept label and system feedback on a falsely selected label.

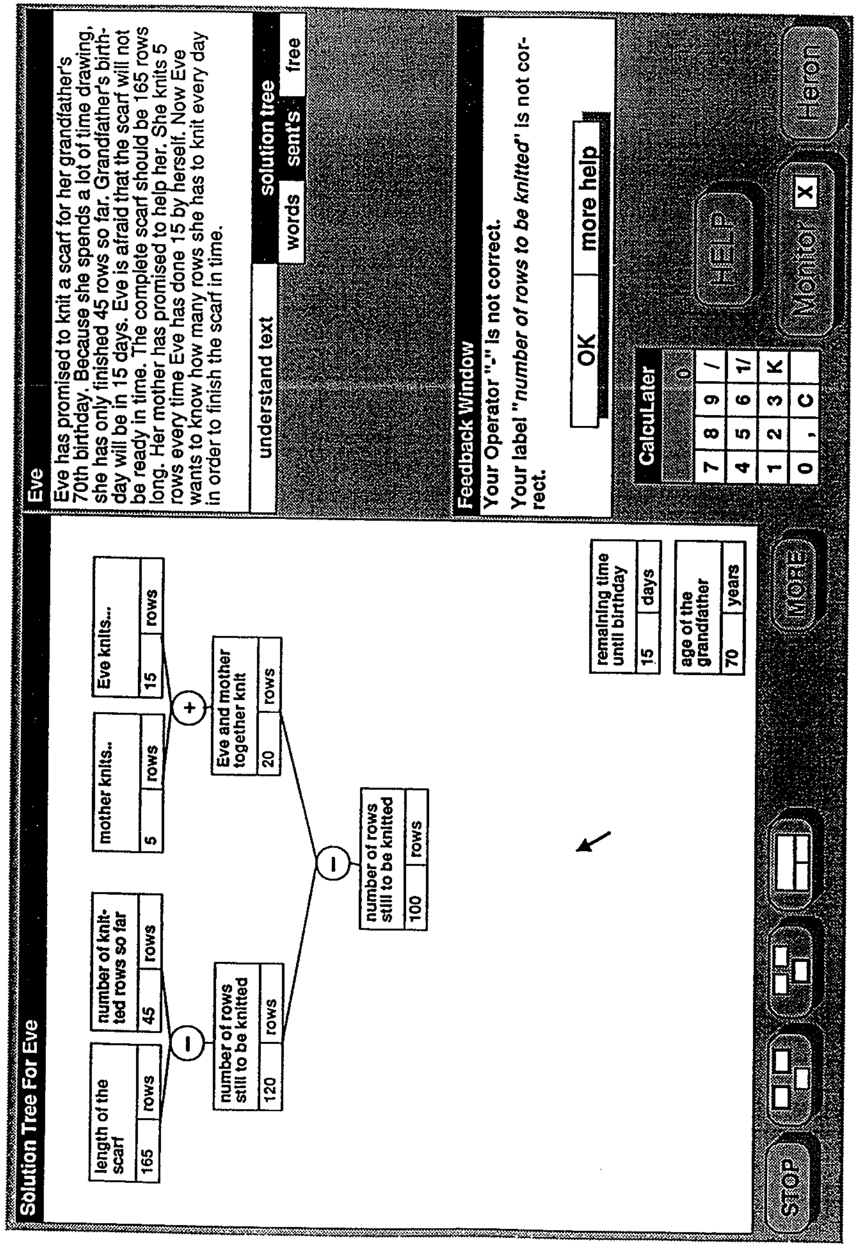
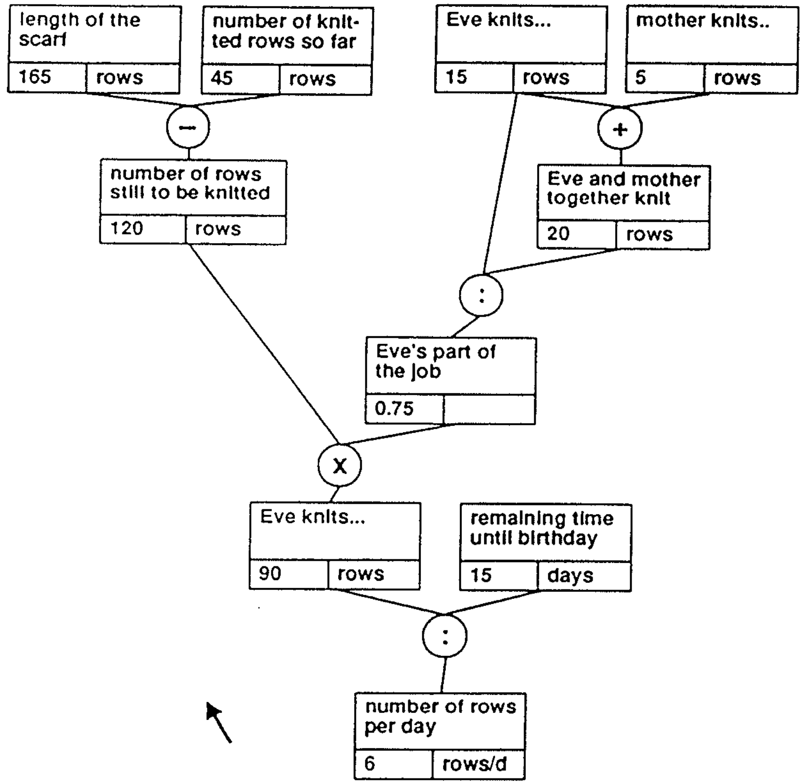


FIG. 5.4. Emerging solution tree; system feedback on an operation error.

empty-operator node. Hereby an empty box, representing an intermediary subgoal unit, is generated by the system. Before the student can place the unit of measurement (*rows*) and label (*number of rows still to be knitted*), into the box, he or she has to select an appropriate mathematical operation from a menu that appears by clicking the mouse on the empty operator node. In our example, the student unfortunately has chosen the faulty situation concept *Eve's part of the job,* causing HERON to intervene with an error message and some (potentially gradual) advice on how to correct the erroneous label (feedback window in Fig. 5.3). Let us assume that the student (after this first message or after having received further—more constraining—advice from the system) has detected his or her error and changed the label to *number of rows still to be knitted,* thus, achieving his or her first subgoal.

Figures 5.4, 5.5, and 5.6 (Fig. 5.4 depicting an operation error made by the student and how the system responds to it) show, for any triadic relational schema being instantiated, how its resulting elements can be used to generate new schemata, that is, to achieve further subgoals by building on the emerging



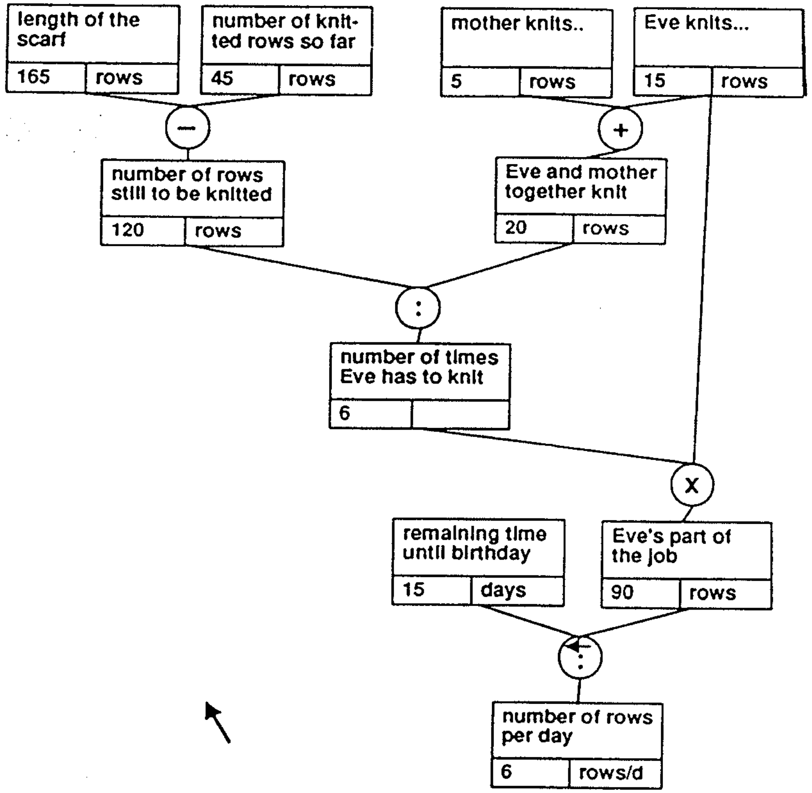


FIG. 5.6. Completed solution tree; alternative comprehension path.

solution-tree network. Solution trees can be constructed through a mixed forward and backward inferencing activity. In forward inferencing, the student's solution starts by constructing triads based on the quantities given in the problem statement; in backward inferencing, the student starts with the goal element, or with some inferred intermediary element, and works backward or upward to the given elements. The tree-constructing activity goes on until the student has completed a tree (Fig. 5.5 or 5.6) and decides that the problem is solved.

It is up to the student to choose which comprehension path to follow, that is, how to navigate through the problem space. Most problems allow more than one comprehension path. Figures 5.5 and 5.6 describe two alternative operative conceptualizations of the same problem.[5]

To generate equations from the solution tree, the student can push the right side of the mouse button on any numerical value in the tree. The system displays

[5]For an example of how a much more complex problem is solved, see Reusser (1993).

a partial equation or, if clicking on the numerical value in final result box, the solution equation of the problem.

Final Remarks

At any time while working on a problem, students can call for progressively constrained help for both deeper comprehension of the problem and construction of solution trees (design principle of *minimal help* or *variable control*). Help with text comprehension mainly consists of explanations of language patterns, paraphrasing of sentences, and hints for textual inferences. Construction help provides the student with increasingly detailed hints about what to do next, ultimately offering a next operative step.

To provide feedback, HERON does not try to model the students' thinking. Feedback is provided on the basis of a detailed analysis of the task space into which the observed construction activities of the students are mapped. HERON knows which situation elements can or must (not) be connected in a solution tree, and which labels and units of measurements are to be attached to quantified situation units. Thus, HERON monitors the students' construction activity and provides feedback on mathematical operation errors, situation-concept errors, unit-of-measurement errors, and errors regarding the inclusion or omission of (ir)relevant information in a solution tree.

Rather than being a solo instrument of learning, HERON is conceived of as a flute in an orchestra (Salomon, 1990), or in a carefully composed learning environment. So far, students mostly work in pairs. HERON records in a log file the activities performed on the screen. This allows students to replay their comprehension (solution) paths, which they can reflect on and discuss (cf. design principle 4). To evaluate the HERON system in a classroom setting, we compared pairs of fifth-grade students solving word problems with and without the system in an intervention study (H. Staub, Stebler, Reusser, & Pauli, 1994). So far, the results indicate that HERON was easily accepted by teachers and students, and that using the system was beneficial to the students in at least three ways: with respect to (a) an improvement in understanding and solving relatively complex story problems in a posttest, (b) the mindfulness of dialogue among the co-working students, and (c) the quality of cooperation in maintaining a mutually shared understanding of the problem (Pauli, 1994).

As with any educational technology, the design, use, and evaluation of an ITS requires a pedagogic-didactic philosophy of both what and how to tutor. That is, it must incorporate reflections about the child as a learner, a view of the pedagogical and cultural goals of schooling, a sound analysis of domain knowledge and tasks, as well as a psychopedagogical theory of development, learning, and instruction.

Like all educational technologies, computer-based learning environments should be considered as a means or an instrument to pedagogical ends or goals.

Computers, which are by their very nature multipurpose tools with many facets, will only become valuable cognitive-didactic tools for learners and teachers if they are designed for and used in the service of pedagogical goals.

ACKNOWLEDGMENTS

The work in this chapter was supported by a grant from the Swiss National Science Foundation (Contract 10–2052.86). I am grateful to the members of my research group (Xander Kämpfer, Fritz C. Staub, Rita Stebler, and Ruedi Stüssi), in which many of the ideas in this chapter were shaped. I am also very grateful to Eileen Kintsch for her invaluable comments on the final draft of this chapter.

REFERENCES

Aebli, H. (1980–1981). *Denken. Das Ordnen des Tuns* [Thinking]. (2 vols.). Stuttgart: Klett.
Aebli, H. (1987). *Zwölf Grundformen des Lehrens* [Twelve basic methods of teaching]. (3rd ed.). Stuttgart: Klett. (Original work published 1961)
Aebli, H., Ruthemann, U., & Staub, F. C. (1986). Sind Regeln des Problemlösens lehrbar? [Can rules of problem-solving be taught?]. *Zeitschrift für Pädagogik, 32,* 617–638.
Anderson, J. A. (1989, April). *Psychology and intelligent tutoring.* Paper presented at the fourth international conference on AI in Education, Amsterdam, Netherlands.
Anderson, J. A., Boyle, C. F., Farrell, R., & Reiser, B. (1984). Cognitive principles in the design of computer tutors. In *Proceedings of the sixth annual conference of the Cognitive Science Society* (pp. 1–9). Boulder: University of Colorado, Institute of Cognitive Science.
Anderson, J. R., Boyle, C. F., Corbett, A. T., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. In W. J. Clancey & E. Soloway (Eds.), *Artificial intelligence and learning environments* (pp. 7–49). Cambridge, MA: MIT Press.
Bobrow, D. G. (1964). *Natural language input for a computer problem solving system.* Unpublished doctoral dissertation, MIT, Cambridge, MA.
Bork, A. (1985). *Personal computers for education.* New York: Harper & Row.
Brown, J. S. (1984). *Idea amplifiers: New kinds of electronic learning environments.* Palo Alto, CA: Xerox Research Center.
Bruner, J. (1986). *Actual minds, possible worlds.* Cambridge, MA: Harvard University Press.
Collins, A., & Brown, J. S. (1988). The computer as a tool for learning through reflection. In H. Mandl & A. Lesgold (Eds.), *Learning issues for intelligent tutoring systems* (pp. 1–18). New York: Springer.
Comenius, J. A. (1659). *Orbis sensualium pictus.* London: J. Kirton.
Cummins, D., Kintsch, W., Reusser, K., & Weimer, R. (1988). The role of understanding in solving word problems. *Cognitive Psychology, 20,* 439–462.
De Corte, E., Verschaffel, L., & de Win, L. (1985). Influence of rewording problems on children's problem representations and solutions. *Journal of Educational Psychology, 77,* 460–470.
Derry, S. J., & Hawkes, L. W. (1989). *Error-driven cognitive apprenticeship: A feasible ITS approach.* Tallahassee: Florida State University Press.
diSessa, A. (1989). *Computational media as a foundation for new learning cultures* (Tech. Rep. No. G5). Berkeley: University of California Press.

Greeno, J. G. (1983). Conceptual entities. In D. Gentner & A. L. Stevens (Eds.), *Mental models* (pp. 227–252). Hillsdale, NJ: Lawrence Erlbaum Associates.
Kämpfer, A. (1991). *Prototypische Entwicklung eines adaptiven tutoriellen Systems zum Verstehen und Lösen mathematischer Textaufgeben* [Development of an adaptive ITS for understanding and solving mathematical word problems]. Unpublished master's thesis, University of Bern, Switzerland.
Kant, I. (1965). *The critique of pure reason* (N. Kemp, Trans.). New York: St. Martin's Press. (Original work published 1781)
Kaput, J. J. (1989). Linking representations in the symbol systems of algebra. In S. Wagner & C. Kieran (Eds.), *Research issues in the learning and teaching of algebra* (pp. 167–194). Reston, VI: The national Council of Teachers of Mathematics.
Kintsch, W., & Greeno, J. G. (1985). Understanding and solving word arithmetic problems. *Psychological Review, 92,* 109–129.
Kintsch, W. (1989, September). *A theory of discourse comprehension: Implications for a tutor for word algebra problems.* Paper presented in the third conference of the European Association for Research on Learning and Instruction EARLI, Madrid, Spain.
Klafki, W. (1958). Didaktische Analyse als Kern der Unterrichtsvorbereitung [Didactic analysis as core of teacher preparation]. *Die Deutsche Schule, 50*(10), 451–471.
Klafki, W. (1963). *Das pädagogische Problem des Elementaren und die Theorie der kategorialen Bildung* [The pedagogical problem of the basics in education]. Weinheim: Beltz.
Lajoie, S., & Derry, S. J. (Eds.). (1993). *Computers as cognitive tools.* Hillsdale, NJ: Lawrence Erlbaum Associates.
Lehmensick, E. (1926). *Die Theorie der formalen Bildung* [Theory of formal discipline]. Göttingen: Vandenhoeck & Ruprecht.
Lepper, M. R., & Gurtner, J.-L. (1989). Children and computers. *American Psychologist, 44*(2), 170–178.
Mandl, H., & Lesgold, A. (Eds.). (1988). *Learning issues for intelligent tutoring systems.* New York: Springer.
Mann, L. (1979). *On the trail of process.* New York: Grune & Stratton.
McArthur, D., Stasz, C., & Zmuidzinas, M. (1990). Tutoring techniques in algebra. *Cognition and Instruction, 7,* 161–195.
Mitman, A., Mergendoller, J., Marchman, V., & Packer, M. (1987). Instruction addressing the components of scientific literacy and its relation to students outcomes. *American Educational Research Journal, 23,* 611–633.
Nathan, M., Kintsch, W., & Young, E. (1992). A theory of algebra word problem comprehension and its implications for the design of learning environments. *Cognition and Instruction, 9*(4), 329–389.
Nesher, P., & Teubal, E. (1975). Verbal cues as interfering factors in verbal problem solving. *Educational Studies in Mathematics, 6,* 41–51.
Ohlsson, S. (1986). Some principles of intelligent tutoring. *Instructional Science, 14,* 293–326.
Oelkers, J. (1989). *Reformpädagogik* [Reform pedagogy]. München: Juventa.
Paige, J. M., & Simon, H. A. (1966). Cognitive processes in solving algebra word problems. In B. Kleinmuntz (Ed.), *Problem solving* (pp. 51–119). New York: Wiley.
Papert, S. (1980). *Mindstorms, Children, computers and powerful ideas.* New York: Basic Books.
Pauli, C. (1994). Mathematische Textaufgahen lösen mit HERON. Eine Analyse von Schüler dialogen in einer kooperativen Lernsituation mit und ohne Unterstützung duich ein tutorielles system [Solving mathematical word problems with HERON. An analysis of dialogues of student pairs cooperatively solving word problems with and without the use of a tutorial system]. Unpublished master's thesis, University of Zurich, Switzerland.
Pea, R. D. (1985). Beyond amplification: Using the computer to reorganize mental functioning. *Educational Psychohlogist, 20,* 167–182.

Pea, R. D. (1987). Cognitive technologies for mathematics education. In A. H. Schoenfeld (Ed.), *Cognitive science and mathematics education* (pp. 89–122). Hillsdale, NJ: Lawrence Erlbaum Associates.

Piaget, J. (1947). *La psychologie de l'intelligence* [Psychology of intelligence]. Paris: Colin.

Piaget, J. (1950). *Introduction à l'épistémologie génétique* [Introduction to genetic epistemology]. Paris: Presses Universitaires de France.

Reusser, K. (1985). *From text to situation to equation* (Tech. Rep. No. 143). Boulder: University of Colorado Press.

Reusser, K. (1989a). *Vom Text zur Situation zur Gleichung. Kognitive Simulation von Sprachverständnis und Mathematisierung beim Lösen von mathematischen Textaufgaben* [From text to situation to equation. Cognitive simulation of language comprehension and mathematization in solving mathematical word problems]. Unpublished habilitation thesis, University of Bern, Switzerland.

Reusser, K. (1989b, September). *Textual and situational factors in mathematical word problems.* Paper presented at the third conference of the European Association for Research on Learning and Instruction EARLI, Madrid, Spain.

Reusser, K. (1990). From text to situation to equation: Cognitive simulation of understanding and solving mathematical word problems. In H. Mandl, E. De Corte, N. Bennett, & H. F. Friedrich (Eds.), *Learning and instruction in an international context* (Vol. 2, pp. 477–498). New York: Pergamon.

Reusser, K. (1993). tutoring systems and pedagogical theory. In S. Lajoie & S. Derry (Eds.), *Computers as cognitive tools* (pp. 143–177). Hillsdale, NJ: Lawrence Erlbaum Associates.

Rousseau, J. J., (1951). *Emile ou de l'éducation* [On education]. Paris: Garnier. (Original work published 1762)

Salomon, G. (1990). Studying the flute and the orchestra. *International Journal of Educational Research, 14*(6), 521–532.

Salomon, G., Perkins, D. N., & Globerson, T. (1991). Partners in cognition: Extending human intelligence with intelligent technologies. *Educational Researcher, 20*(3), 2–9.

Scardamalia, M., Bereiter, C., McLean, R. S., Swallow, J., & Woodruff, E. (1989). Computer-supported intentional learning environments. *Journal of Educational Computing Research, 5,* 51–68.

Shute, V., & Glaser, R. (1990). A large scale evaluation of an intelligent discovery world: Smithtown. *Interactive Learning Environments, 1,* 59–77.

Simon, H. A. (1981). *The sciences of the artificial.* Cambridge, MA: MIT Press.

Sleeman, D., & Brown, J. S. (1982). *Intelligent tutoring systems.* New York: Academic Press.

Staub, F. C., & Reusser, K. (1995). The role of presentational structures in understanding and solving mathematical word problems. In C. A. Weaver, S. Mannes, & C. R. Fletcher (Eds.), *Discourse comprehension. Essays in honor of Walter Kintsch* (pp. 285–305). Hillsdale, NJ: Lawrence Erlbaum Associates.

Staub, F. C., Stebler, R., Reusser, K., & Pauli, C. (1994, April). *Improving understanding and solving of math story problems through collaborative use of a computer tool (HERON).* Paper presented at the annual meeting of the American Educational Research Association AERA, New Orleans.

Stern, E. (1991, April). *The role of language in solving word problems.* Paper presented at the annual meeting of the American Educational Research Association AERA, Chicago, IL.

Stern, E., & Lehrndorfer, A. (1992). The role of situational context in solving word problems. *Cognitive Development, 7,* 259–268.

Stüssi, R. (1991). *WATGRAF. Ein tutorielles System zum Lösen von handlungsbezogenen Sachaufgaben* [A tutorial system for solving mathematical word problems]. Unpublished master's thesis, University of Bern, Switzerland.

van Dijk, T., & Kintsch, W. (1983). *Strategies of discourse comprehension.* New York: Academic Press.

Weniger, E. (1952). *Didaktik als Bildungslehre* [Didactics as theory of bildung]. Weinheim: Beltz.

Wertheimer, M. (1945). *Productive thinking.* New York: Harper & Row.

Willmann, O. (1957). *Didaktik als Bildungslehre* [Didactics as theory of bildung]. Freiburg: Herder. (Original work published 1889)

Wittmann, E. C. (1992). Mathematikdidaktik als "design science" [Didactics of mathematics as design science]. *Journal Mathematikdidaktik, 13*(1), 55–70.

Lawrence Erlbaum Associates, Inc., Publishers
10 Industrial Avenue
Mahwah, New Jersey 07430

Library of Congress Cataloging-in-Publication Data

International perspectives on the design of technology-supported learning environments / edited by Stella Vosniadou . . . [et al.].
p. cm.
Includes bibliographical references and indexes.
ISBN 0-8058-1854-5 (cloth : alk. paper). — ISBN 0-8058-1853-7 (paper : alk. paper)
1. Instructional systems—Design. 2. Computer-assisted instruction. 3. Educational technology. I. Vosniadou, Stella.
LB1028.38.I59 1996
371.3'078—dc20 95-47474
CIP

Books published by Lawrence Erlbaum Associates are printed on acid-free paper, and their bindings are chosen for strength and durability.

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

International Perspectives on the Design of Technology-Supported Learning Environments

Edited by

STELLA VOSNIADOU
University of Athens
ERIK DE CORTE
University of Leuven
ROBERT GLASER
University of Pittsburgh
HEINZ MANDL
University of Munich

LAWRENCE ERLBAUM ASSOCIATES, PUBLISHERS
1996 Mahwah, New Jersey