A

The Impact of Teaching Expertise on Educational Software Selection: An Examination of the Strategies Used by Teachers and Novices in their Approach to Software Selection

Susan Spier Krauss

A dissertation submitted to the Graduate Faculty in Educational Psychology in partial fulfillment of the requirements for the degree of Doctor of Philosophy, The City University of New York

2000

UMI Number: 9969703

UMI

UMI Microform 9969703

Bell & Howell Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

This manuscript has been read and accepted for the Graduate Faculty in Educational Psychology in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

Date 5/1/2000 Chair of Examining Committee

Date 5/1/2000 Executive Officer

Dr. David Rindskopf

Dr. Linnea Ehri

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

Abstract

The Impact of Teaching Expertise on Educational Software Selection: An Examination of the Strategies Used by Teachers and Novices in their Approach to Software Selection.

by

Susan Spier Krauss

Advisor: Professor Carol Kehr Tittle

Expertise in teaching has been associated with a comprehensive knowledge base, well organized schemas resulting from a deep understanding of the problem, pattern recognition skill, the ability to appropriately apply abstract concepts to concrete situations, and the ability to recognize features of the problem central to the solution (Borko & Livingston, 1989; Leinhardt and Greeno, 1986; Sabers, Cushing, & Berliner, 1991.) The introduction of computer technology to the classroom has added, for some teachers, an unfamiliar dimension to the classroom environment, a dimension in which their problem solving expertise may not be as effective. This study examines the impact of computer technology on teachers' approaches to the problem of evaluating educational software packages for instructional merit.

Sixteen teachers and 14 novices evaluated two educational software packages for educational merit. Two of the teachers had expertise in educational technology. The remaining teachers and the novices had no formal training in using educational technology. Participants' "think-aloud" responses were recorded, by audio and video tape, as they evaluated the software, and their responses to a brief interview and survey were collected.

Teachers generated a greater percentage of technical and pedagogical statements, but did not differ significantly from Novices in their attention to specific Pedagogical variables. A qualitative analysis revealed that teachers and novices had different approaches to the problem solving task directed in part by schemas they held for effective instruction. Further, the technology-trained teachers appeared to have greater access to their schemas for effective instruction than those teachers for whom the computer was an unfamiliar environment. These findings suggest that technology training may need to be an integral part of teacher education programs.

Acknowledgements

This project was made possible through the efforts of many people.

I would like to express my gratitude to Dr. Carol Tittle, my advisor, for her support, guidance, and inspiration throughout the process. Dr. Tittle encouraged me from the very beginning of my doctoral studies and provided me with the confidence and enthusiasm I needed to keep working toward the goal. I feel fortunate to have had the opportunity to be her student.

I also want to express my appreciation to Dr. David Rindskopf and Dr. Linnea Ehri for their participation on this project and throughout my studies, and Dr. Burt Flugman and Dr. Hope Hartman for their suggestions and help.

Thank you to all the people who participated in the study, and the family and friends who helped in so many ways.

A very special thank you to my Mom, Joyce M. Spier, and my brother, Ira J. Spier, for believing in me and supporting me in this endeavor. Their help was invaluable.

And one for Ethan and Gabriel, my beautiful sons, who patiently spent countless hours watching me work.

Finally, to my wonderful husband Michael, for making sure this dream came true for me. I would not have completed my studies if not for him. This is his dissertation too.

This project is dedicated to the memory of my father, Howard E. Spier.

TABLE OF CONTENTS

LIST OF APPENDICES

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

Introduction

The differences in the behavior and mental processes of experts and novices have intrigued researchers across domains. Observable and inferred characteristics of the individuals, and the products of their processes, help researchers to identify people as either experts or novices, or to place them at some other point along this continuum.

Researchers have focused on problem solving as a means of examining the differences in the cognitive processes of experts and novices. Expertise within a given domain is associated with effective problem solving skills within that domain (Johnson, 1988; Lawrence, 1988; Voss & Post, 1988). Analysis of problem solving strategies provides a window into the mental processes on which experts and novices may differ (Chi, Feltovich, & Glaser, 1981; Glaser, 1984; Larkin, McDermott, Simon, & Simon, 1980).

Expertise has been associated with several characteristics; the possession of a comprehensive knowledge base; well organized schemas that facilitate the application of higher order, abstract concepts to problem situations; superior pattern recognition performance; automaticity of task performance; and the ability to discriminate between pertinent and superficial features of a problem (Glaser & Chi, 1988; Greeno, 1976). The quality of the individual's representation of the problem has also been shown to be related to expertise (Chi, Feltovich, & Glaser, 1981). The expert-novice paradigm has had strong implications for researchers in education for it attempts to describe and explain the distinction between experts and novice teachers, thus establishing a framework through which researchers can address the integral questions of how and why these distinctions are affected.

Teaching, a complex, cognitive skill (Borko & Livingston, 1989) is comprised of a multitude of tasks or problems. These problems, embedded in the context of content and pedagogically related issues, (Berliner, 1991) are manifested in the larger, more global objective of teacher effectiveness. Teachers must have a deep understanding of the structure, concepts, and relationships between concepts for a given domain, yet they must also be able to translate that knowledge into knowledge that is accessible to their students. The degree to which teachers succeed in conveying given knowledge to their students, often measured through student achievement outcomes, has been used as an index of teacher expertise (Shulman, 1986).

Expertise in teaching has been associated with a variety of teaching behaviors, essentially approaches to the problem of how to most effectively convey information to students. In a general sense, expertise in teaching mirrors many of the patterns found in expertise in other domains. Expert teachers are able to recognize patterns quickly and to automate routine tasks thus freeing up cognitive processing resources for more challenging situations (Kagan, 1992). Expert teachers also have well organized, comprehensive knowledge bases that comprise their schemas for teaching (Borko & Livingston, 1989; Kagan, 1992; Leinhardt & Greeno, 1986).

Berliner (1991) identified three knowledge structures on which the tasks that constitute teaching may be organized. The first of such structures is termed script, and is defined by a teacher's knowledge of everyday teaching experience, such as checking homework, or presenting new information. The second, termed scene, describes a teacher's knowledge of the people and objects in classroom events, such as reading groups, or independent seatwork. The third, termed propositional, is characterized by a teacher's knowledge of students, of the subject matter, and of given pedagogical strategies. The selection of materials for use in a lesson (a problem, or task, in which most teachers engage) exemplifies how a teaching task can be described by the three knowledge structures. Selection of materials requires that teachers can effectively identify how the materials can be used in their scripts; how the materials can be used in their classrooms; and how their students, as learners, will interact with the materials, in the context of the subject matter.

The evaluation and selection of microcomputer instructional software can be understood as a problem which teachers electing to use computers as instructional tools within their classrooms may encounter. The goal of software evaluation typically represents a teacher's desire to identify a software package which will best serve the teacher's instructional aims for his or her students (Schueckler & Shuell, 1989). Since teachers must often select software without the benefit of testing it on their students, they are, in essence, making a hypothesis about the effectiveness of the software based on the knowledge that they have about the subject matter, their students, and their classroom environment. This situation raises research questions about how teachers represent this problem, in particular whether there are differences between experts and novices.

Kagan (1992) summarizes studies demonstrating that expert teachers and novice teachers have different ways of representing the problem of teaching, as experts and novices in other domains have different representations for problems. These findings have had interesting implications for research on teachers' evaluation of software. Does the representation of the problem of what constitutes sound instruction that expert teachers possess allow them to more effectively evaluate instructional media than novices, who may not possess this representation? Research suggests experts are able to distinguish between superficial aspects of a problem and integral information (Greeno, 1976). Will expert and novice teachers differ on the elements they recognize in the software as serving a critical role with respect to its instructional effectiveness? Will expert and novice teachers differ on the elements they recognize in the software as serving a superficial role with respect to the goal of instruction, even when these superficial elements take the form of engaging graphics and sounds? Is teaching expertise a prerequisite for the skills necessary to identify instructionally effective software? Is knowledge of educational technology as well as knowledge of teaching necessary to identify instructionally effective software?

The present study explores these questions. A group of novices, people without teaching experience or background, a group of experienced elementary school teachers, and a group of teachers with educational technology training and experience were asked to evaluate the instructional effectiveness of two mathematics software package for elementary school students.

Participants in the study were asked to talk aloud while using the software packages. Their responses were audio and video taped, transcribed, and then analyzed.
The objective of the analysis is to provide a window into the cognitive processes engaged in by the experienced teacher who is expert in educational technology, experienced teachers without the technology expertise, and rank novices, when they are faced with the problem of selecting instructionally effective software.

The ability to evaluate the effectiveness of instruction is one of the attributes of expertise in teaching (Kagan, 1992). As computer technology continues to play a more significant role in the classroom, and the number of instructional software packages continues to grow, it becomes increasingly important for educational researchers to understand the problem solving strategies expert teachers invoke when evaluating the effectiveness of educational software. How teachers expert in educational technology and teachers without this expertise evaluate software will provide educational researchers with a tool that can be used in furthering teacher professional development courses in the area of educational technology.

CHAPTER II

Expert Novice Paradigm and Teaching

What Is Problem Solving?

Gagne (1966, p. 132) defines problem solving as "...an inferred change in human capability that results in the acquisition of a generalizable rule which is novel to the individual, which cannot have been established by direct recall, and which can manifest itself in applicability to the solution of a class of problems." Problem solving involves an interaction of the problem solver with the problem. This interaction is associated with a problem solver's awareness of the stimuli relevant to the problem, directions, or information from external sources directing a problem solver's attention, and instructions, or information about the solution desired (Gagne, 1966). The process of solving the problem is strongly associated with the instructions, which not only define the desired solution but assist the solver in identification of relevant aspects of the stimulus, recollection of appropriate rules, and guidance of the thinking process (Gagne, 1966). The process can be summarized as 1) the stating of the problem, 2) the defining of the problem, 3) the searching and formulating of hypotheses, and 4) the verification of the solution (Gagne, 1966).

A problem can be described as a situation in which one state exists, but another state is desired (Newell & Simon, 1972; Voss & Post, 1988). For example, the state of a long division example may consist of a divisor and a dividend, but not the desired quotient. The problem is thus the solving of the example, or, th finding of the quotient. The solved problem is represented by the new state, the example with its quotient.

Problems are not limited to well-structured domains like mathematics. Consider the example of a man who has moved into a one bedroom apartment but must accommodate the furniture he previously had in a three bedroom house. The initial state is a truck full of furniture; the goal state is a comfortable arrangement of that furniture in his new home. The problem is to arrange the furniture.

The aforementioned scenarios illustrate problems that appear on opposite ends of a continuum that spans well-structured to ill-structured problems. Well structured problems are those with a limited number of open constraints, problems for which very few conditions will satisfy the solution (Reitman, 1965). The long division problem is a well structured problem. Clearly, the long division problem has but one solution. Within the community, there will be little dissent over the feasibility of the solution. However, the ill-structured, furniture-arranging problem may have a multitude of possible solutions. Although certain proposals might be obviously rejected, such as the bed in front of the entrance way, there are a considerable number of workable solutions that would be accepted by the problem solving the community.

For both well and ill-structured problems, successful problem solving has been associated with the problem solver's expertise in the problem domain (Glaser & Chi, 1988). Researchers' understanding of expertise has increased significantly in the past twenty years.

What Is Expertise?

While the causal agents of expertise may not be able to be definitively determined, researchers have identified behaviors associated with expertise, and have inferred from these behaviors the nature of the experts' cognitive processes.

Research suggests that while general heuristics may be associated with effective problem solving, expertise appears to be related to the problem solver's organization of domain specific knowledge (Glaser, 1984, Glaser & Chi, 1988). Expertise is believed to be associated with the individual's knowledge base and the organization of the schema that comprise the knowledge base (Gagne, 1966). Experts have well connected, well organized schema. Novices may have information that could be useful in the solving of a problem, but tend to have less well organized schema, and as a result, may not as easily access the information they need.

Experts are also able to recognize meaningful patterns in their own domains (Glaser & Chi, 1988). It can be argued that this pattern recognition skill is a critical factor in experts' superior ability to accurately represent problems, as they are able to note the characteristics attributable to given problem categories.

The way in which the problem is represented by the problem solver is the mediating factor between the structure of the knowledge base and the problem solving process. Glaser (1984) suggests that "[t]he quality, completeness, and coherence of this internal representation determine the efficiency and accuracy of further thinking" (Glaser, p. 98). That is, the solver's representation of the problem will determine the information, both declarative and procedural, and the strategies he or she will search for in long term memory. It becomes clear the extent to which the problem representation and knowledge base structure are related. A poor representation of the problem will set the stage for faulty solving strategies, but without a well organized, comprehensive knowledge base, a problem solver will not be able to recognize the patterns that serve to let him or her accurately represent the problem.

Poor problem representation and a superficial understanding of the problem domain may be attributable to novices' tendency to be attentive to the overt, yet superficial components of a problem, while overlooking more integral characteristics. Novices often do not have a deep understanding of the problem situation. In contrast, experts' deep understanding of the domain, allows them to appropriately apply theoretical principles. Novices often cannot apply such principles, even when they are aware of them; in other words, they fail to make the connection between the concrete, specific pieces of information and the more abstract, global principles. Finally, experts have superior self-monitoring skills; they spend more time than novices on the analysis of the problem and are better able to judge the difficulty of a problem, and to know when they need to check themselves (Glaser & Chi, 1988). These characteristics of experts are summarized in Table 1.

It is not inadequacies in the processing capabilities of novices, but rather, limitations in their knowledge bases that characterize their performance. What follows is a discussion of the difference between experts' and novices' approaches to problem solving.

Table 1

Behavioral and Cognitive Characteristics of Experts

Characteristics
Well organized, comprehensive knowledge base
Excellent pattern recognition skills
Metacognitive skills
Ability to apply theoretical concepts to problems
Ability to distinguish relevant from superficial information
Ability to chunk units of information into larger units
Ability to automate routine tasks freeing up cognitive capacities

Experts and Novices in Well-Structured Problem Domains: Models of the Problem Solving Process

The Role of a Comprehensive Knowledge Base in Effective Pattern Recognition

Many models that attempt to describe the process of problem solving, and the variables involved in that process, have been proposed. Several of the models use the concept of a production system. Simon (1975) defines a production system as "an ordered set of processes called productions" (p. 274). Each production contains a condition and an action. If the condition of a production is found, its action will be performed. One could argue that a production system is basically a collection of `If-Then' statements, analogous to the "conditions" and "actions." For example, presence of the condition "if one side of the triangles is shared," executes the action "then congruent sides are present." More complex cases involve chains of productions, with a given production's execution contingent upon the features of a previous one.

Greeno's (1976) research suggests that pattern recognition might be a key element in the problem solving process. Using a computer simulation, he postulates a problem solving process, using as an illustration, the problem of proving two triangles congruent.

The computer is initially provided with a data structure, essentially a database, of the information provided by the problem. In the case of proving two triangles congruent, the program is given both the physical attributes of the triangles, and the goal, that being to prove the triangles congruent. The problem solving procedures, including the rules of what constitutes congruency, comprise a production system that is utilized in the solving of the problem.

Greeno's (1976) model is organized around three categories of knowledge: pattern recognition knowledge, knowledge of propositions for inference, and strategic knowledge. Pattern recognition essentially involves the locating of a set of features and relations in the data structure, which when identified, will execute the production.

Inferential productions are actually propositions (in this case, geometric rules, such as "vertical angles are congruent.") If vertical angles are identified (via the execution of the requisite productions) the action of this production adds to the database the structure (or fact) that the triangle has congruent angles.

The strategic knowledge component of the process involves goal and subgoal setting and the verification that goals have been met. Greeno differentiates between definite goals, those for which a pattern can be matched by a single production, and indefinite goals, those for which the execution of more than one production could satisfy the goal. In other words, there is more than one way to prove congruency of a triangle; therefore, the proving of congruency is an indefinite goal.

In sum, Greeno's computer program is equipped with knowledge of the criteria that must be met for two triangles to be congruent. In this case, since multiple criteria will satisfy the goal, this well-structured problem has an indefinite goal. The program scans for patterns that match those in the database of features that contribute to the attainment of the subgoals and/or goals. The program has instructions (in the form of strategic productions) to continue running until the goal has been met, or presumably, until it can be determined that the goal cannot be met.

Greeno's simulation does not address the differences between expert and novice problem solvers, but it does identify variables that could be construed as influential in determining an individual's level of expertise. If the initial database is lacking information, or the information is poorly organized, perhaps due to inexperience in a particular domain, pattern recognition will be impacted negatively. That is, subjects may note that the triangles share a side, but this fact may be passed over as irrelevant, or, they may not attend to it at all.

In fact, an ill-equipped database will negatively impact the strategic productions as well. It will become more difficult to establish subgoals if a real understanding of congruency is not present.

As another illustration, consider the problem of trouble-shooting a malfunctioning computer. Assume the computer boots successfully but fails in many of the applications. The expert trouble-shooter has a very large database of productions. He or she may formulate the goal of fixing the computer, with one of the subgoals that of getting Microsoft Windows to work properly, and another subgoal of assuring there are no viruses present. The expert has to know how Windows should run. He or she must also know that a virus could be responsible for the problem, and how to identify the presence of a virus. Presumably this expert's long term memory contains a database of literally hundreds of other pieces of declarative and procedural information. The successful trouble-shooter recognizes the patterns of computer behavior that might be associated with certain circumstances, or to use Greeno's model, through pattern recognition the trouble-shooter determines that a condition exists and an action is executed.

However, without a well organized, comprehensive database, the computer technician is not as likely to be successful. Even if the problem solver knows the components that comprise a solution, inability to recognize patterns will prevent productions from being executed. In other words, not knowing that the configuration of jumper switches on a motherboard will impact the running of the machine translates into the setting of the jumpers going unrecognized as a vital piece of information. Thus, the novice technician does not solve the problem.

Problem Representation

How a problem is represented by the solver has been found to be related to the problem solving processes engaged in by the solver (Chi, Feltovich, & Glaser, 1981). Problem representation is a fundamental component of problem solving (Glaser, 1984). Chi, Feltovich, and Glaser (1981) define a problem representation as "a cognitive structure constructed corresponding to a problem, constructed by a solver on the basis of his domain-related knowledge and its organization" (p. 121). Essentially, when a solver encounters a problem, he or she must determine the type of problem that it is, and this determination will guide the process of problem solving. Researchers have proposed several explanations of how this determination is made. The domain of physics was used to formulate the problem solving models proposed by Chi, Feltovich, and Glaser (1981) and McDermott and Larkin (as cited in Larkin at al, 1980).

McDermott and Larkin describe four stages of problem representation. Stage one involves a literal representation of the problem statement. This stage includes the identification of relevant keywords in the problem statement.

Stage two, termed the "naive" stage, involves identifying the literal objects and their spatial relationships as stated in the problem and a sketch of the situation. An individual with minimal experience in the domain would be capable of achieving this stage.

Stage three is comprised of scientific representation of the problem using idealized objects and physical concepts such as forces and energies. During this stage the equations needed to solve the problem are generated.

Stage four contains the algebraic representations that result from the equations produced in stage three.

An alternative model of problem solving representation using physics as the domain was proposed by Chi, Feltovich, and Glaser (1981). Their model differs from the McDermott-Larkin model in the degree to which there is interaction among the stages. Problem representation is seen as the result of the bottom-up processes of categorizing the problem and the top-down processes of testing that categorization against the knowledge base available to the solver.

The difference between the expert's and novice's approaches to problem solving were examined using four studies (described in Chi, Feltovich, and Glaser (1981) to identify the differences in outcomes between experts and novices, and the differences in the cognitive processes and structures that experts and novices engage when solving physics problems.

Studies one through three focused on the role of problem categorization in problem solving. The research question was to examine the relationship between problem categorization and attempts at solution. The initial study involved subjects (eight physics doctoral students as experts and eight undergraduates who had completed a semester of mechanics as novices) formulating categories for 24 physics problems based on their similarity of solution. The results of the study found no differences in the number of categories produced by experts and novices; however, experts seemed to categorize problems according to the laws of physics while novices relied more on keywords or surface features of the problem.

Experts took 18 minutes to complete the task; novices were finished in 12 minutes. It was inferred that this disparity could be explained by the fact that the experts were processing the information on a deeper level than the novices.

To research this further, a second study was developed which more closely compared the features examined by the novices and the experts. A collection of problems was assembled that had surface features crossed with applicable physics laws. As expected, the experts were more focused on underlying principles while the novices concentrated on the surface features of the problems as their basis for classification of the problems.

In a third study, experts and novices were given category labels generated by experts and asked to tell everything they could about problems involving each of the category labels, including how these problems might be solved. The rationale for this study was to allow the researchers to more closely examine the schematas associated with the category labels, that is, the schematas that are accessed via the category labels.

The experts' protocols suggested that their schematas contained basic physics principles, whereas basic physics principles were not initially mentioned by the novices. It appears that the experts associate physics principles with procedural knowledge, and have a strong sense of their applicability. Experts' production rules, essentially "if-then" conditions for action, also contain actions that contained explicit solution methods. The novices' production rules did not contain actions that could be characterized as explicit solution procedures. Their actions were more characteristic of attempts to find specific unknowns, that is, to fill in unknown variables.

The final study analyzed the basic approaches to problem solving taken by experts and novices. Experts were able to use physics principles to guide their approaches. They began with a hypothesis or a few hypotheses, and tested them against the features of the problem. Novices, more conscious of surface features, formulated surface-oriented categorizations that yielded associated equations, and their focus was on solving for variables, rather than the global problem.

In sum, the evidence presented in these studies of problem representation in physics suggests that experts and novices categorize problems differently, with experts' categories eliciting a schema that functions in the representation of the problem and includes potential solution methods. Experts are able to derive from problem statements features that activate these schemas. Novices, in contrast, search for keywords or surface features in the problem statements, and then attempt to solve for unknown variables. They do not access schemas that include possible solution methods based on a representation of the problem as related to laws of physics.

This suggests that novices do not have as rich an understanding of the laws of physics as the experts; they are not able to recognize the applicability of the laws in hypothesized solutions. It could be argued that their understanding is of a more superficial nature; they view the problem as a series of equations that need to be solved, but do not recognize how the laws of physics are applied in practical situations.

Novices tend to take analogous approaches to problem solving in other domains. The author has observed novices (college students in a developmental math class) calculating percentages of whole numbers. They do not rely on mathematical principles, but follow a series of steps that they believe will yield a solution, much like the novices in the physics problems did when they initially attempted to solve for unknown variables before they analyzed the problem in terms of what laws of physics would be appropriate. Even when the developmental math students proposed answers that were illogical, for example, 27% of the whole number is larger than the whole number, they did not recognize the logical flaw. It could be argued that they really did not understand what a percentage was. Rather, they had procedural knowledge that was not firmly linked with any concept.

Schemas and Information Access

The way in which information is stored in long term memory appears to be a significant contributor in explaining the differences in performance between experts and novices. Larkin et al (1980) identify a number of components that impact on an individual's level of expertise: perceptual knowledge, recognition capabilities, and the way in which information is represented in long-term memory.

A now famous experiment during which master chess players and novices were asked to duplicate the configuration of pieces on a chessboard sheds light on the role of perceptual abilities in expertise (Chase & Simon, 1973). While chess players demonstrated superior ability over novices to recreate the board when the configuration matched a possible game, there was no difference between the performance of experts and novices when the pieces were arranged randomly on the board. These finds lead researchers to conclude that the perceptual abilities of the experts did not exceed those of the novices, but that another phenomenon was operating. The experts were able to chunk the information, that is, to group many chess piece positions as one chunk or one configuration in memory because they were able to recognize certain patterns that were part of the chess game. These patterns would not be familiar to the novices, which would account for their inadequacy in duplicating the board. However, when the pieces were placed haphazardly on the board, the recognizable chess piece configurations or patterns did not exist, leaving the experts without that advantage. This translated into their being no better than the novices at duplicating the board. The ability to recognize patterns was linked to expertise in the particular domain.

Larkin et al (1980) hypothesize that a large set of perceptual patterns serve as an index, or access route, to the expert's factual and procedural knowledge. Through pattern recognition, experts access factual information and information about actions and strategies that may be appropriate in the context in which the pattern exists. Long-term memory, according to this hypothesis, is comprised of a large set of productions. A production consists of a condition and action. If a stimulus satisfies a condition, the corresponding action is evoked. For example, an individual may see a plus sign, a stimulus which satisfies a condition and is accompanied by a given action. Actions may be the recovery from memory of internal representations of external information.

Larkin et al (1980) propose that human memory is comprised of a complex organization of nodes connected by links, termed a list structure. Objects and elements of objects correspond to nodes. Relations between objects correspond to links.

With a set of physics problems to be solved as the task, Larkin et al compared the difference between expert and novice problem solving strategies. Experts completed the task in one quarter of the time needed by novices and with fewer errors. Think aloud protocols and worksheets provided additional information for the researchers on the strategies employed by the experts and novices.

Novices solved most of the problems by working backward. Presumably this means that they focused on the solution required and possible routes (equations that should be solved) that would lead to the solution. Experts worked forward for the easiest problems. It was inferred that they had enough experience with these problems to simply solve all the equations, a process which would lead them to a full understanding of the problem and would yield a solution. Their strategy seemed to be to accumulate information about the initially unknown quantities in order to solve the problem. Novices, inexperienced with the material, seemed to require goals and subgoals to direct their progress. The productions of the expert could be characterized as the condition "if there is knowledge of the independent variables" leading to the action "solve for the dependent variable." In contrast, the production of the novice contained the condition "if the dependent variable is the desired quantity" with an action of "solve the equation."

The major difference between the productions of the expert and novice, seen in the protocols, was that the actions of experts were to solve the problem whereas the actions of the novices were to create a goal to solve the problem. Perhaps this difference lies in the automaticity of the expert's procedures. The expert links many steps in the process as one; the many steps are executed without deliberation. It appears that the novice must think out the steps in the procedure, a circumstance which may account for the greater time required to complete tasks and increased demands on cognitive processes.

Strategies In Well-Structured Problems

One defining component of expertise is the ability to select appropriate strategies, to recognize when the strategy should be abandoned, and to recognize when the strategy should be continued. Expert problem solvers typically have access to effective strategies. Experts' superior metacognitive facility coupled with a comprehensive knowledge base allows them to monitor their progress and make decisions concerning the effectiveness of the strategy they've selected (Alexander, 1992; Glaser & Chi, 1988).

The interesting question of what influences strategy selection may best be answered by analyzing the cognitive demands of the different strategies and the level of expertise of the problem solver. Novices may not have the knowledge bases, nor experience, to be able to select the most appropriate strategy (Alexander, 1992).

Even well structured problems, those for which only one outcome will satisfy the solution, may be solved using a variety of strategies. Simon (1975) examines four strategies which may be employed in the solving of the well-structured problem, the Towers of Hanoi puzzle (see Figure 1). The puzzle requires the solver to move all of the disks that are initially placed on one peg to another peg. The puzzle contains three pegs. One constraint placed on the solver is that at no time may a larger disk be placed upon a smaller one.

Figure 1. The Towers of Hanoi.

The first strategy examined, the goal-recursion strategy, requires the subject to establish goals, meet them, and reestablish goals until the puzzle is solved. In the case of a three disk puzzle, the subject might represent the goal as three individual subgoals, those being the movement of the clusters of disks from the pegs. These goals serve as guides for the operations performed, and the solver does not need to refer to the distribution of disks on the pegs. The solver does not even need to see the puzzle if he or she can retain the unaccomplished part of the goal in memory.

The second strategy, a perceptual strategy, requires the solver to think of the problem as rebuilding the stack or pyramid onto another peg. A disk can be moved on to a peg if there are no smaller disks resting on it, and no smaller disks on the target peg. To avoid creating configurations whereby it becomes impossible to achieve the goal, the solver must retain a stack of subgoals in short term memory. A third, more sophisticated perceptual strategy requires the solver to identify the largest disk blocking the movement of the disk the solver desires to move.

The fourth strategy, a move-pattern strategy, employs rote procedures to complete the task, in which the solver moves the pegs in a given sequence without focusing on the perceptual elements of the problem. Odd numbered moves require one set of steps; even numbered moves require another, and only the parity information needs to be stored in short term memory.

These strategies can be classified as either reasoned or rote; the move-pattern strategy is clearly rote in that the pattern of moves is learned and the demands on short term memory are minimal. The solver is not required to scan for certain configurations or to store the perceived image in short term memory.

Simon developed a program that functions as a model of human problem solving in certain task environments, specifically those in which a given situation, the problem situation, can be compared with the goal situation to discover what differences exist between them, with the goal or goals of eliminating these differences. The General Problem Solver contains operators associated with each difference that can be applied to eliminate the difference; if no conditions exist for which the operator can be applied, then the current situation is compared with a situation for which conditions would exist to determine what those differences would be, and if there exists an operator that will eliminate these differences, it is applied.

This model is the basis for the goal recursive and perceptual strategies. Essentially, those strategies test for the legality of a given move determined to be useful in facilitating the solution. If the move is legal, it is made; if not, the solver considers alternative actions that will eventually allow for that move to be made. The General Problem Solver also tests for the legality of the move it determines will serve toward eliminating differences between the current state and the goal state. If the move is not legal (if operators do not exist that will eliminate the difference between the current state and the desired state) the goal of rearranging the current situation so that the desired move is legal becomes the targeted problem.

Some interesting questions are raised as a result of this discussion. Although the impact of expertise on strategy choice was not discussed, it could be argued that only someone with a fair amount of experience with the problem could abstract out the rules used in the rote strategy. Although presumably novices could be taught the rules, they would not be able to use them in altered conditions of the problem (the rote strategy does not transfer from a four disk problem to a five disk problem.) The goal recursion strategy requires that solvers have an understanding of the concept of recursion, and that they have the ability to recognize a new situation as similar to the starting situation so they could perform the recursion. Thus, the solver would have to have a knowledge base that contained information organized in a way that would allow for him or her to recognize a similar problem or problem situation. This would hold true for the perceptual strategies too.

What can be concluded from the discussion in Simon (1975) is that there are many strategies available for solving problems even as structured as the Tower of Hanoi problem. The unanswered research questions are how individuals select a strategy and whether there is interaction between strategy choice and level of expertise. It has been suggested that expertise is indeed a factor in strategy selection (Alexander, 1992).

Experts and Novices in Ill-Structured Problem Domains

Making Decisions

Expert-novice research has traditionally focused on problem solving approaches in highly structured domains such as physics and mathematics. Ill-structured domains present different challenges for problem solvers. While most math problems can be satisfied by only one solution, a variety of solutions often satisfy an ill-structured problem.

For example, consider the problem of writing a short story. While there may be constraints, such as an approximate number of pages, the fact that characters should retain a certain degree of consistency throughout the story, and that the plot should follow a logical sequence, an almost infinite number of solutions could satisfy the problem of creating a short story. In fact, even within the constraints, a wide variety of conditions would be satisfactory. The story could be 40 pages or 50 pages. It could contain three characters or nine characters. It could be set anywhere. It could have an upbeat ending or a depressing conclusion. When the problem of creating a story is compared with the problem of finding a solution to an algebra problem, the distinctions between ill- and well- structured problems become apparent.

Ill-structured problems, those typically found in the social sciences, political science, and psychology can be characterized as problems which have loosely-defined initial and goal states (Reitman, 1965). Problems can be characterized as ill- or well structured according to the number of constraints that require resolution, with ill-structured problems requiring the resolution of a large number of constraints and being problems for which a variety of solutions will satisfy the constraints (Reitman, 1965; Simon, 1975).

Simon (1975) contends that ill-structured problems may become well-structured problems in the hands of the problem solver. That is, the solver represents the ill-structured problem into several well-structured problems.

It could be argued that this ability is a measure of expertise, as experts in even an ill-structured domain will have had enough experience to recognize open constraints that in reality are closed. For example, the novice dress designer might believe there are an infinite number of ways to design sleeves, but the expert has the experience to rule out most of the options, thus reducing the number of values that can be placed in the open constraint. Figures 2 and 3 provide examples of the types of constraints found in well-structured problems - closed constraints, and ill-structured problems - open constraints.

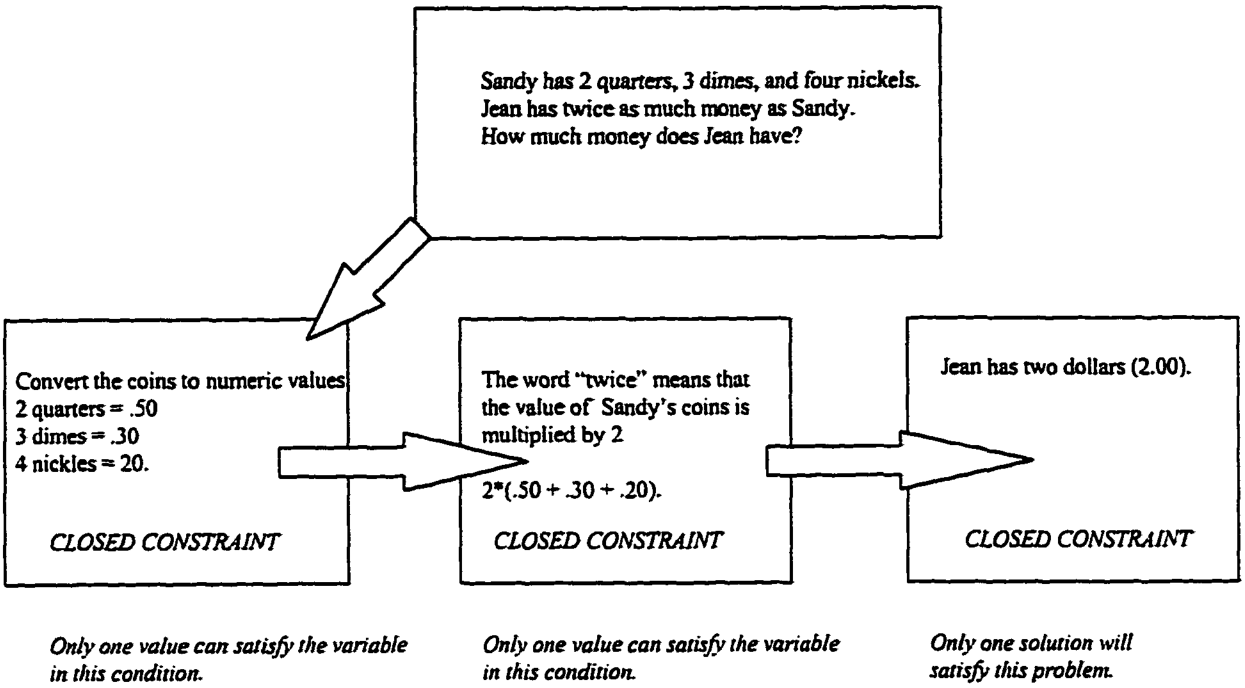

Figure 2. Diagram of a Well Structured Problem

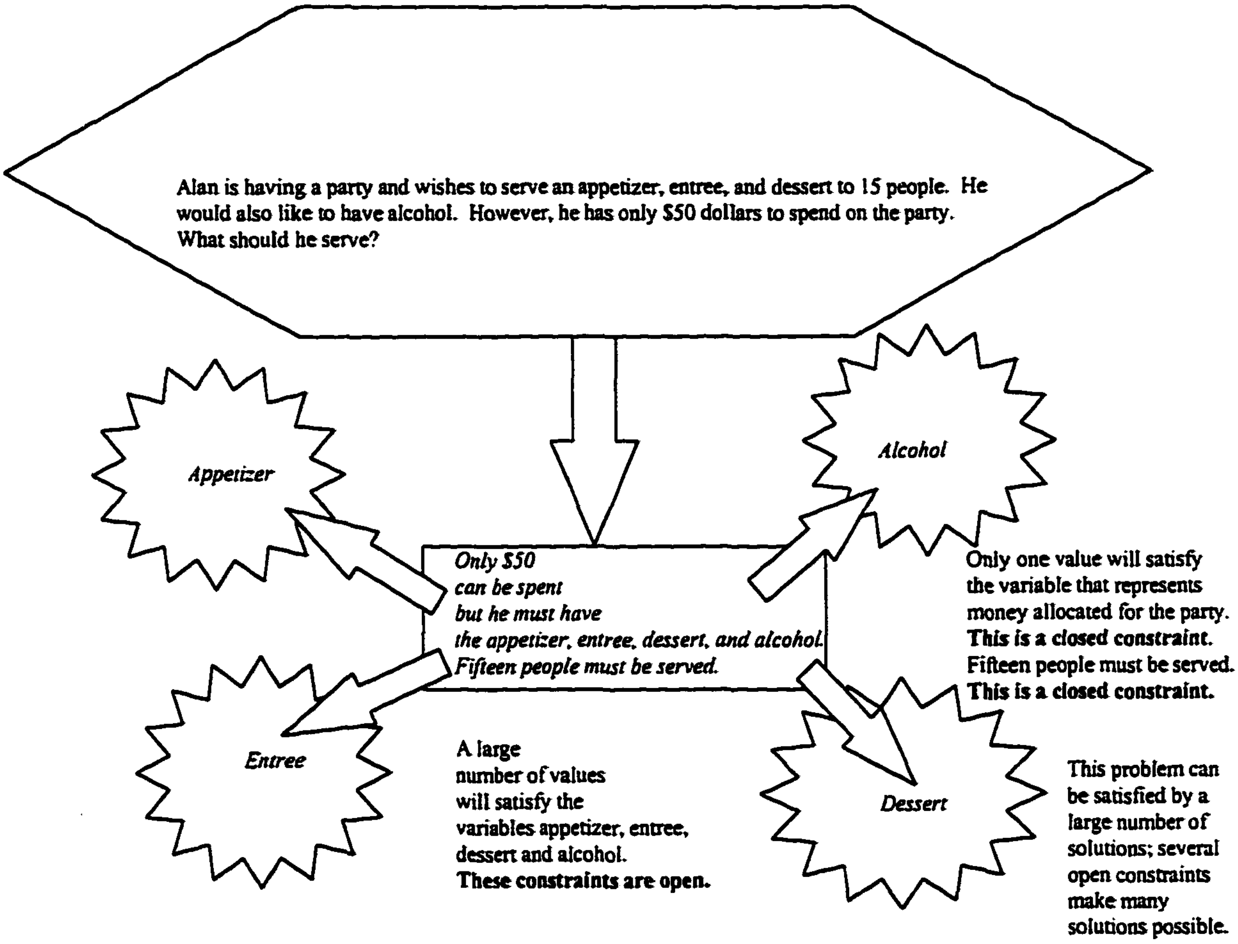

Figure 3. Diagram of an Ill-Structured Problem

Planning

All problem solving requires some planning. Hayes-Roth and Hayes-Roth (1979) define planning as "the predetermination of a course of action aimed at achieving some goal" (p. 275). It can be argued that the ill-structured problem may necessitate more complex planning than the well structured problem due to the variety of options that would satisfy the solution.

Although planning has been traditionally conceptualized as having a top-down, hierarchical structure, Hayes-Roth and Hayes-Roth (1979) propose a model suggesting that planning follows an opportunistic structure. Throughout both the planning and execution of the plan, the planner's decisions and observations affect future decisions with respect to the plan in ways that may or may not follow a hierarchical framework. Planners select options at various points in the temporal sequence and with different guiding principles at levels of abstraction.

The model is comprised of several variables that serve to categorize intentions and actions of the planner. The blackboard functions as a desktop where five conceptual planes, plan, plan-abstraction, knowledge base, executive decision, and meta-plan, are recorded.

The plan plane has four levels of abstraction and consists of actions the planner intends to take. 1) The outcomes phase is the most abstract level, and indicates what the planner intends to accomplish by executing the plan. Hayes-Roth and Hayes-Roth (1979) gave a subject the task of accomplishing a list of errands in a given time frame, an ill-structured problem to the extent that the subject would have to decide which errands were to be accomplished, as it was impossible to accomplish all the errands in the time allotted. The outcome of the subject's plan would be to accomplish the errands. 2) The designs level characterizes the behavior the planner intends to take to achieve the outcome, i.e., the decision to drive toward the southeast part of town. 3) The procedures level specifies specific actions, such as which errand in that part of town to do first. 4) The operations level describes more specific actions, such as which streets would be used to go from one errand to another.

The plan-abstraction plane describes attributes of potential decisions. The attributes represent general characteristics of decisions without specifying any particular decision. For example, decisions might be spatially motivated; the planner might decide to do the errands that are close together in sequence. A specific area of town is not mentioned, just the fact that a decision to organize the plan around spatial considerations is relevant under the plan-abstraction plane. The plan-abstraction plane contains four levels characterizing decisions proposed in the corresponding level of the plan plane. There is the intention level, in which the planner, for example, decides which errands he or she intends to accomplish; the scheme level that suggests generating a design incorporating spatial clusters of errands; a strategy level which indicates that the closest errand be selected; and a tactic level to search for short cuts between errands. The levels follow a hierarchy, with the intention level leading to the scheme level leading to the strategy and tactic levels.

The knowledge base plane is comprised of observations and information that might affect the planning process. The knowledge base is situation specific so the model uses the names of the given situation to label the levels of abstraction. The errand level refers to the weighting of the importance of each errand. The layout level refers to the observation a planner might make with respect to the locations of the errands. The neighbor level would refer to the observation a planner might make that two errands are near each other, and finally, the route level would contain a short cut that the planner observed.

The executive decision and meta-plan planes determine features of the planning process in contrast to the plan, plan-abstraction, and knowledge base which determine features of the developing plan.

The model can be applied to a variety of situations. A teacher who wishes to present a science lesson on the solar system may have, as an outcome, the goal of having the students be able to name all nine planets, know their relative proximities from the sun, and the conditions of the planet, including temperature. At the designs level, is the teacher's decision to show a video on the solar system. The procedures level might include each student being responsible for taking notes on a given planet during the video. The operations level might encompass the more specific actions of assigning the planets to the students.

The plan abstraction plane for such a scenario might be the decision to actively engage all the students during the presentation so that nobody is daydreaming once the television is on. The knowledge base plane might contain observations such as the fact that the lights are producing glare on the television screen, prompting the teacher to shut off the lights.

One of the strengths of this model is its attentiveness to the dynamic quality and flexibility of successful plans. Planners make changes to the plan as the planner deems necessary. It could be argued that experts, in contrast with novices, can more easily identify when changes are necessary, and more successfully affect them.

Decision Making in an Ill-Structured Domain

Decision making is a form of problem solving. When the decision is to chose which peg to move in order to most expediently complete the Towers of Hanoi problem, the possible solutions are limited. However, ill-structured problems present a problem solver with a sometimes inexhaustive set of options or decisions.

Judgments can be conceptualized as tasks made for the purpose of choosing between alternative actions (Einhorn, 1980). Shulman and Elstein (1975) cite Newell (1968) in their discussion of what comprises judgment. They consider judgment to be "...one set of admissible responses; where classes or categories are given, it is usually called selection, estimation, or classification (Shulman & Elstein, 1975, p. 15). They distinguish between "preparation for judgment," which may be extended through time with subprocesses, and "judgment," which they refer to as an immediate process, as "...distinguished from searching, discovering, or creating, as well as from musing, browsing, or idly observing" (Shulman & Eldstein, 1975, p. 15). Several models of the judgment process have been proposed (Newell in Shulman & Eldstein, 1975). Models with Maximum Formal Generality are based on the proposition that decision making can be represented by formal, mathematical approaches, such as the Bayesian or regression models. The goal of the regression model is to determine the weighting scheme that best represents the judge's processing of information. The Bayesian model focuses on the impact of new information on the judges' beliefs.

Models of Task Environment attempt to model the major distinctive features of a task environment. These approaches are not general and do not rely on judgmental law.

Models of the Information Process attempt to explain the relationship between the data used in the judgment and the laws that govern the judgment. These models are often expressed in terms of computer programs. They aim to provide an explanation of human thought processes.

Research suggests that decision makers attempt to reduce the complexity of their tasks by attempting to reduce the number of alternatives, thus simplifying the task (Payne, 1980). The decision maker stores information about the alternatives and compares the attributes of the alternatives with their schema of which attributes are important or relevant to solving the problem. The attributes have been assigned values, and these values are compared for the given attributes to eliminate alternatives, the goal being the elimination of all alternatives but one (Payne, 1980).

In selecting from a set of alternatives, a standard or set of criteria must be generated or accepted, upon which alternatives can be evaluated (Einhorn, 1980). Essentially, decision makers perform around the constraints of the following set of equations, where x is the variable or attribute under observation, and c is the criterion cutoff point (Einhorn, 1980).

if $x >= x_c$, take action A;

if $x < x_c$, take action B.

The successfulness of the decision maker, however, is difficult to measure, as only the outcomes based on the selected alternative are evaluated. That is, whether or not the selected alternative A is successful, the success or failure of rejected alternative B is not determined. Furthermore, research suggests that people tend to focus on the frequency of positive outcomes, ignoring false positives, false negatives, and negative outcomes. That is, for example, a manager hiring employees must select 10 people from a pool of 100. She may use the criteria of GPA as a factor in her decision making process, and select only those applicants who scored above a 3.7. If seven out of 10 of the employees perform to her satisfaction, she will tend to focus on this fact, ignoring: the false positives, those employees who were accepted due to their GPAs but are not good workers; the false negatives, those candidates rejected as employees because of their GPAs but who would have been outstanding workers; and the negative "hits," those who were rejected and would have performed poorly.

Unlike laboratory conditions, real world situations usually do not allow decision makers to test alternatives before rejecting them. In addition, decision makers in real world situations do not typically look for disconfirming evidence to reject established rules when making decisions. This is particularly significant in the area of educational software selection; teachers tend not to look for disconfirming evidence when deciding on the merit of a software package. That is, they are not as likely to search for a situation where color and sound are detrimental to achieving learning outcomes as they are to search for those instances where they support the outcome. This can be evidenced by the large proportion of software evaluation forms which contain checklists of factors associated with positive learning outcomes. Those packages which meet the proposed criteria are accepted; those that do not are rejected. The instances of positive student response serve to support the processes used in the making of the decision.

Lawrence (1988) poses a model that suggests the process experts use to arrive at decisions. Using legal decision making as the domain, the model was used to analyze the decision making processes of experienced and novice magistrates. The model is based on the premises that a frame of reference, that is, the individual's prior experiences and external constraints, impacts information selection. The selection of information affects inferences and judgement. An If Then design illustrates the relationships between these variables. If a certain frame of reference is present and certain external constraints are present, then the individual will select certain information. For example, a very experienced magistrate might know exactly which information is important and select accordingly, given the availability of the information, an external constraint. The information selected will then be used in drawing inferences about the situation and be instrumental in arriving at a decision.

To test the model, Lawrence (1988) used three subjects, two experienced magistrates, and a novice who had completed all his legal training and service as a clerk of the court and was currently working as a clerk and chamber magistrate giving informal legal advice to the public. Sentences were to be passed by the subjects on three cases selected from the court archives. The first case involved a 51 year old homeless man who pleaded guilty to a charge of driving with more than the higher prescribed limit of alcohol. Case two involved a middle eastern migrant who pleaded guilty to stealing 14 dollars worth of goods from a large city store although she had the money to pay for them with her. The third case involved a young banker who pleaded guilty to charges of supplying and possessing Indian Hemp.

Subjects were recorded using a talk-aloud procedure and the transcripts were coded into the following basic concepts: sentencing objectives, views of offenses, information about offenses, information about defendants, inferences, and sentencing decisions. The following patterns emerged: The experts' structure included the objective of using treatment to avoid reoffense. Using the model to analyze responses, it follows that if the objective is to avoid reoffense and the view of the crime is either serious or nonserious, then the subject will select events, legal and personal details according to patterns, objectives and individualized treatment, and then the subject will infer guilt, reasons, problems, or prospects, and then sentence for treatment or to avoid reoffense.

For the novice's structure, if the objective is to impose a tariff or parity sentence, and the view of the crime is either serious or nonserious, then the novice will select events, legal and personal details according to norms and single features, and then will infer if there were any mitigating circumstances, and then will sentence for tariff or treatment, if details indicate mitigation.

The powerful role that frame of reference plays becomes apparent on comparison of the two structures. View of the crime as serious or not serious will lead to given events being selected; initial objectives, the goal of the subject, will also impact on the data that is selected, and of course, inferences are made on the data that is selected, inferences which lead to a decision. The goal of the experts, their initial frame of reference, was that they didn't want reoffense; this served to structure much of the processes that lead to their final decisions. The goal of the novice was to find the appropriate tariff or parity sentence, a goal which also strongly influenced what data would be selected and what inferences would be made.

Lawrence (1988) attributes the differences between the experts and the novices to familiarity with the cases, patterns that are consistent for common offenses, and experience with the procedures for interpreting details. These attributions are not unlike those made to explain differences between experts and novices in well-structured domains, as the strategies used by experts and novices in ill-structured domains parallel the strategies used by experts and novices in well-structured domains. Pattern recognition ability, familiarity with the task, and experience interpreting details have all been shown to distinguish experts from novices in well-structured domains. The decision maker possesses a schema for the domain, with experts' schemas more sophisticated and well organized than the schemas of novices (Carroll, 1980).

Johnson's (1988) study of experts and novices in ill-structured domains also suggests several commonalities shared by both well- and ill-structured domains. In both the well- and ill-structured domains, experts concentrate on a few features or variables whereas novices attend to significantly more, possibly contributing to the differences in time needed to complete the tasks. Experts also complete tasks more quickly than novices (Johnson, 1988).

Johnson (1988) relied on verbal protocols as a window into the cognitive processes that experts used in an ill-structured decision making task. Two studies were conducted through which the differences in both process and performance could be examined. In the first study, physicians on the admissions committee that matches applicants with residencies and internships at teaching hospitals, and undergraduate novices, were asked to rate applicants on a five-point scale. Each of 200 applicants had a folder which contained about 13 pages of information, including standardized test scores, letter of application, letters from the deal and faculty of the applicant's medical school, transcripts, and summaries of interviews with applicant.

Verbal reports were obtained from two physicians and two undergraduate novices as they reviewed six of the applications; overall ratings of all twelve physicians and a single novice were taken for 156 applications. Subjects' statements were coded as follows:

"Retrievals	Nonevaluative statements consisting of verbatim or paraphrased quotation of information presented in the folders.
Recall	Statements of similar nonevaluative information obtained from memory.
Evaluation	Statements which result in the judgement of some aspect of the applicant, his or her medical school or other object. This excludes judgments made in response to a question on the response form, or evaluation read from the folder.
Scaling statements	Responses made in completing the forms provided with each application.
Inferences	Nonevaluative statements, based

	on retrieved information, but which clearly go beyond the presented information.
Goal statements	Statements of intentions or actions to be performed. Search for a source of information, etc.
Miscomprehension	Statements reporting difficulty in understanding presented information.
Comment	An uncodable statement or one irrelevant to the task" (Johnson, 1988, p. 216).

Some patterns emerged from the findings, suggesting differences in the processes but not products of the experts and novices. Times of task completion for experts were about 7.8 minutes compared with 15 minutes for novices, possibly because experts examined a smaller amount of information. The novices examined 43 percent of all the statements that were available in the folder, whereas the experts examined about 22 percent. The novices and experts also examined different information, seeming to reflect experts' beliefs that some of the information was not as valuable. Experts also had different searching strategies, jumping around in the folders, whereas novices tended to proceed through the folder in a linear fashion. Experts also tended to focus on relatively rare variables, applicable to the case under consideration. The novices' responses contained twice as many retrievals as the experts'. Experts' responses included more goal statements than novices (3 percent to 1 percent).

The second study yielded similar findings. The experts for this study were financial analysts; the novices were business administration students. Their task was to predict year end closing prices for 40 securities.

The securities were described using 22 variables typical of those available to analysts predicting security prices. Summaries of Wall Street Journal news stories accompanied half of the securities, but the 40 securities were not named so that the subjects' ratings would not be based on information that was recalled from the news stories.

Differences between experts and novices followed those of the first study. Experts performed faster than novices, averaging about 144 seconds to novices' 162 seconds per security. Experts examined less information than novices, focusing on fewer variables and different variables. Twenty-two percent of all information examined by experts could be accounted for by two variables, earnings per share and the previous year's closing price. These variables accounted for only five percent of the information examined by novices. Novices examined 21 out of 22 variables whereas experts confined themselves to only 13.

Interestingly, performance did not appear to be influenced by process differences for either study. Experts and novices achieved essentially the same level of performance. In the second study, experts slightly outperformed the novices, possibly because the news information was more helpful to the experts than the novices. Thus, both studies suggest that in ill-structured domains, differences in the processes of experts and novices far exceed differences in their performance.

Johnson (1988) suggests that in ill-structured domains, experts do not outperform the simple regression model, possibly because experts may place too great a weight on certain rare events, neglecting base-rate data. The cognitive demands necessary to mentally create a simple regression model, one that properly weights base-rate data, may exceed that of the decision maker's capability, thus offering an explanation for the fact that the regression model outperforms the expert.

Decision making may involve judging the degree to which variables covary, that is, the absence or presence of a relationship as well as the strength of that relationship. The accuracy of naive perceivers' judgment of covariation appears to be related to individuals' schemas for the scenario, qualitative features of the variables, and characteristics of the decision task itself (Crocker 1981). Crocker describes a five step model of the cognitive processes involved in making covariation judgments.

Step one involves assessing which data are relevant to the covariation judgment. Confirming instances tend to be regarded as relevant whereas disconfirming cases are more likely to be ignored.

Step two involves the sampling of cases or instances of the variables under question. The collection of data used to support an assertion may suffer sample bias; individuals select instances that are familiar to them but do not include those instances that are not as easily accessible.

Step three involves classifying the instances as confirming or disconfirming cases. Expectations about a given situation may influence how individuals perceive relationships between variables (Crocker, 1981). The categorization of ambiguous cases, those instances that do not clearly fall into one level of a variable or another, has been found to be subject to the perceiver's expectations. This finding suggests, for example, that a belief that use of educational software leads to increased academic success may bias an individual's assessment of covariation between use of a particular software package and the academic success of a class. A software package that is not clearly instructionally sound may be classified as a disconfirming instance of the assertion that the use of educational software is associated with higher grades. Additionally, it has also been found that instances that are disconfirming cases are often discredited as unreliable evidence, even to the extent of disqualifying a large sample as unrepresentative of the population at large.

Step four involves recalling the evidence and estimating the frequencies of confirming and disconfirming cases. Expected instances are more likely to be recalled, and are more likely to have an advantage in frequency estimation.

Finally, step five involves integrating the evidence into an assertion.

Crocker (1981) does not examine the difference in the accuracy of covariation judgment between novices and experts in a given domain; that is, she does not consider the effect of familiarity in an area on performance in this task. For example, an expert horse trainer may be better able to assess the relationship between practice runs and wins than a novice.

The Role of Schemata in Decision Making

Ill-structured domains pose interesting questions regarding the role of schemata in problem solving or decision making. Wineburg (1991) suggests that in domains such as history, the demands of typical tasks faced by expert historians cannot be met simply through the use of "problem templates" or principle-oriented knowledge structures. The problems that historians solve depart from those of experts in other domains, such as physics for example, in that physicists are usually faced with facts from which they must formulate a solution, whereas historians begin with an outcome or solution, and derive what led up to this outcome. This departure may represent a domain for which expertise is not necessarily associated with highly developed schemas for which a broad base of declarative knowledge is essential.

The differences in the approaches that expert and novice historians used when asked to evaluate texts and paintings for accuracy was examined in a study by Wineburg (1991). Subjects included eight historians, six of whom held doctorates--the remaining two were in doctoral programs, and high school students about 16 years of age. The experts came from a variety of specializations; represented were specializations in American Education, Western United States, American Business, Native Americans, Japan, 17th Century England, British Social History, and Medieval Islam. The subjects were directed to read aloud seven documents, which included two diary entries, an excerpt from an autobiography, a formal deposition, a newspaper report, a letter of protest, a selection from a historical novel, and an excerpt from a high school textbook. The subjects were asked to comment on the texts as they were reading and then rank the documents in the order of their trustworthiness. Another part of the task required subjects to rank three paintings according to how accurately they portrayed what happened on Lexington Green. Subjects were also given a test of their knowledge of 12 terms relevant to the Colonial period.

Four categories of protocols were established for the painting analysis task. Responses were classified as <u>description</u> if they included descriptive references that had no bearing on the purpose or function of the feature being described. Responses were classified as <u>reference</u> if they included statements that referred back to the written documents or paintings for the purpose of corroboration. Responses were categorized as <u>analysis</u> if they included reference to the goals, aims, or objectives of the author or artist. Responses were labeled <u>qualification</u> if they included commentary that qualified other statements, for example, adding reservations about a comment or conclusion would be classified as qualification. All other statements were categorized as miscellaneous and not analyzed in this study.

Three categories of responses, considered heuristics, for the document analysis task were used in coding. Subjects were engaged in corroboration if they referred to other documents; sourcing occurred if subjects checked the source of the text prior to reading it; and contextualization referred to instances where the subject attempted to situate a document in the correct temporal and spatial context.

The results of the study suggest marked difference in the performance and cognitive processes of the experts and novices. Experts exceeded novices in performance on the identification of terms task, with mean of 1.8 of the terms accurately identified by novices, and 7.1 for experts. Experts produced considerably more statements during the picture evaluation task than novices (M=52 compared with M=28.) There were significant differences in the amount of analysis, reference, and qualification categories. While the number of descriptive comments was not significantly different for the two groups, experts' responses tended to describe features that had a bearing on the historical aspect of the painting more often than did students.

In selecting the painting that most accurately portrayed the events, four historians agreed on one painting, the one least chosen by the students. Historians' selections came with qualifications. Students seemed to base their selection on the quality of the artwork, paying special attention to realism and detail. It could be argued that this is reminiscent of the finding of expert-novice studies in more structured domains such as physics, where novices seemed more focused on surface features of a problem (Chi, et al 1981). It would seem that the novices in this domain are also focusing on surface features, in this case, the surface features of the painting.

Seven out of eight historians placed the paintings in correct chronological order; only one of the students did. It is not clear from the study whether historians had direct declarative information that enabled them to complete this task or they were relying on inference.

Students and historians also differed on their ranking of the documents. All eight historians chose one of the diaries as the most or second-most trustworthy document; no student ranked the diary that high. All but one of the historians ranked the textbook last or second to last. Three out of eight students ranked the textbook as the first or second most trustworthy.

Historians used corroboration more frequently than did students, with the historians rejecting some statements as true based on the discorroborations from other more reliable texts. Historians used sourcing 98 percent of the time, compared with the students' 31 percent. All eight historians read the attribution at least once before reading the document. Only three of the students did this. Historians used the sources heuristic to develop hypotheses and construct meaning. They regarded all texts as biased, but strived to understand how the bias was operating and what it potentially meant in terms of the reliability and accuracy of the text. Students tended to see bias as a binary variable; either a text was biased or it was not. They saw texts as conveyors of information; historians saw them as the voices of people. For students, the textbook was conceived of as a straightforward account of historical events. The historians had a different schema for textbooks with respect to their biases and accuracy, noting the general tendency for them to be patriotic and to often gloss over details.

Finally, experts were able to rely on their own personal experiences and knowledge of the world to infer how a historical situation might have impacted on the feelings of those present at the time, affording them a greater understanding of the motivations of the people involved in the historical event, and affording them a more realistic view of what probably occurred.

Wineburg (1991) argues that expertise in history does not lie in what the experts know, but rather in the knowledge they have that allows them to "determine the validity of competing truths" (p. 84). One could argue that experts do in fact have a schema, but it does not necessarily have to contain a broad base of facts (many of these experts were really novices in the domain of Colonial history). Rather, a schema may consist of a comprehensive set of procedures not dependent on any specific scenario in order to be useful. It could also be argued that the departure from the schemas used by experts in less ill-structured problems may not be as great as Wineburg appears to suggest. The expert computer programmer is typically able to learn unfamiliar languages because he or she has a general schema for how best to learn the languages, knowing in a general way, which facets of the new language are likely to be important, such as syntax. The syntax of the new language may be markedly different from the one he or she knows, but the expert can recognize that a syntax of some kind does exist, and then has strategies for determining how it works. This is not so very different from the historian who knows in a general way that civil wars are typically caused by given reasons.

It appears that it is the representation of the problem that differentiates the experts from the novices. For the novices, the problem has an answer that does not necessarily have to be inferred, but rather, one that is at once obvious and incontrovertible, hence their lack of qualifications. That is, the novices believe that they do not have to draw inferences to find the solution, but rather that they can rely on declarative information they possess or that appears in the documents.

It could be hypothesized that expert and novice teachers evaluating the effectiveness of a science lesson might fare the same way. Expert teachers would probably have a sense of whether the lesson was going well, even if they didn't have a rich background in the subject matter. For example, they would draw inferences about how effective the instruction was by watching the reactions of the students. Novices might not be able to do so, probably because they wouldn't recognize the patterns that indicated that students were bored, restless, or otherwise disengaged from the lesson. Likewise, expert teachers evaluating a software package would probably be able to draw inferences about its effectiveness even if they were not that familiar with educational software. They would be able to, more easily than novices, identify those elements in the software that would suggest effective instruction, due to their representation of the problem.

Summary

Research suggests that experts have expansive, well organized knowledge bases, that they have superior pattern recognition skills within a domain, that they can readily discern information relevant to a problem, and that they have highly developed metacognitive ability (Chi, Feltovich, & Glaser, 1981; Glaser, 1984; Larkin et al, 1980). These characteristics appear to be associated with expertise across domains, both well and ill-structured (Glaser, 1984, Glaser & Chi, 1988).

While ill-structured domains are characterized by their relatively broad base of open constraints and possible solutions, their position on the ill-structured/well structured continuum may be a function of the representation of the problem by the solver. For example, solvers may represent an ill-structured problem as a series of well structured problems (Simon, 1975).

Teaching has been conceptualized as an ill-structured domain. Like other ill-structured domains, a larger number of conditions may satisfy a solution, and a greater number of constraints remain opened. Researchers have examined the differences between expert and novice teachers with the objectives of identifying the characteristics of teaching expertise, and understanding the cognitive processes associated with those characteristics (Sabers, Cushing, & Berliner, 1991). The following section examines the differences between expert and novice teachers.

Teaching as an Ill-Structured Decision-Making Problem

Teaching has been conceptualized as a "process by which teachers make reasonable judgments and decisions with the intent of optimizing student outcomes" (Shavelson & Stern, 1981, p. 471). In making decisions, teachers are choosing from a pool of instructional techniques those approaches they judge to be appropriate according to the teachers' interpretation of the situation. The process involves the identification and organization of a large amount of information. This information, information about students, the classroom and school environment, and the content of the lesson, is interpreted by teachers. It is the teachers' interpretation of these stimuli that guides or influences their decision making (Shavelson & Stern, 1981), or, as it could be restated, their representation of the problem.

Information is selected and organized, in part, on the basis of a few heuristics and according to attributions for the causes of effects. These heuristics include selection information for salience, judgment of the probability of an event, classification of persons or objects, and revision of initial judgments (Shavelson &

Stern, 1981). The problem solving framework thus exists from which teachers can make manageable the overwhelming amount of information that would ordinarily overload their cognitive capacities.

Experts and novices exhibit different behaviors, behaviors which suggest that they possess qualitatively different cognitive processes with respect to the variables associated with teaching. For example, experts and novices have different schemata for lesson planning and delivery, for assessment of their students' affective and cognitive states, for classroom management techniques, for prioritizing their goals. These schemata are shaped by such variables as prior knowledge, belief systems, and experience. In addition, experts and novices differ on strategies used to attain goals, and on goals they set (Leinhardt & Greeno, 1986). Through observation and other forms of data collection (policy capturing, lens modeling, process tracing, simulated recall, case study, and ethnography) (Shavelson & Stern, 1981) in connection with the teaching process, researchers have attempted to identify the many variables that contribute to the teaching process. They then formulate conclusions about the differences between experts and novices on these variables, with the goal of formulating an understanding of what is happening when novices develop into experts.

Teaching is comprised of a multitude of activities ranging from classroom-management oriented tasks like taking attendance, to learning-oriented tasks like planning what analogies will be appropriate to convey a given concept. A variety of models have been developed that function to organize the activities that teachers perform into categories or variables. Since the difference between experts and novices across domains appears to be related to the knowledge they have and how this knowledge is organized, what follows is a description of some models of teaching, and where the differences between experts and novices lie within these models.

Teachers' Knowledge Bases

Berliner (1991) targeted three sources of knowledge for the competent performance of teaching tasks: content knowledge, pedagogical content knowledge, and pedagogical knowledge.

Content knowledge is defined as "a teacher's understanding of the structure, salient concepts, relations among concepts, and ways of thinking that are characteristic of such curriculum areas as history, physics, and English literature" (p. 147). Thus, given that teaching expertise is related to subject area knowledge, an expert math teacher may not be an expert

English teacher, even if all of his or her general teaching heuristics are in place. Subject expertise is necessary (though not sufficient) for teaching expertise, as extensive understanding of a subject provides the teacher with a clearer picture of how to present material. Inadequate understanding of the material may pose obstacles to making priority decisions about content coverage (Borko & Livingston, 1989). That is, lack of expertise in the subject area often results in the inability to make decisions about what content is important to include and what can be omitted. Novices reported not knowing enough about their subject area to make such decisions (Borko & Livingston, 1989).

Having expertise in a domain may not make for the ability to convey that knowledge. In other words, experts in a given domain may not be expert teachers. They may be lacking pedagogical content knowledge, defined as a "teacher's ability to transform content knowledge into the forms needed to be learned by ordinary students. This would be appropriate metaphors, analogies, simplifications, conceptual webs, relations for everyday life, forms of practice, and transfer opportunities that many teachers of content use to promote learning" (Berliner, 1991, p. 147). Pedagogical content knowledge then marries subject matter knowledge with knowledge about ways to convey subject matter knowledge. It becomes the distinction between having expertise in a field, and having expertise and in addition, being able to convey that information to others. Sabers, Cushing, and Berliner's (1991) study comparing the teaching abilities of experienced teachers of science with scientists who had no teaching experience supports the role of pedagogical content knowledge.

Subjects in this study were classified as either experts, advanced beginners, or novices. Experts were selected from a pool of junior and senior high school science teachers. Fifty-five experts identified by superintendents and principals were then evaluated by project personnel knowledgeable about research in teaching with either classroom teaching experience or training in classroom observation techniques. Those teachers receiving highest ratings from the first observer were visited by a second observer. Seventeen teachers were selected as experts. A third observation was made of these teachers resulting in seven teachers being selected as the experts for the study.

Advanced beginners were selected from a volunteer pool of secondary science student teachers or first year classroom teachers. Their student teaching records were reviewed to ensure that they were competent and possessed the potential to develop into excellent teachers. Four were selected.

Five novices were selected from a pool of 25 individuals with experience in business or industry in the fields of chemistry, engineering, or computer technology. They had no formal pedagogical training.

The purpose of the study was to identify differences in the way experts, advanced beginners, and novices perceived and interpreted classroom events. Subjects were asked to simultaneously view three videos showing different views of a single classroom, and to report on what they observed in the videos. Differences in perception and interpretation of events may account for differences between experts' and novices' teaching practices.

One class period of a junior high science classroom was video taped for a week, with the conditions of the classroom unchanged. Limited classroom space and the obtrusive nature of three cameras necessitated use of a single tape edited into three tapes, each showing a different view of the classroom. The video tape was selected for opportunity to view interaction with the teacher and students, presentation of scientific content, and involvement of the students in the lesson. When the tape was played, each monitor showed approximately one-third of the room, thus, the viewer needed to be simultaneously aware of all screens in order to experience the activity of the classroom at that moment.

The lengths of the three tapes varied according to how well the edited portions could be fit together without the participants noticing any editing effects. The video for the middle monitor lasted approximately 25 minutes and was used to begin and end the viewing.

The left monitor started about three minutes after the middle began and lasted about 12 minutes. Five minutes after the first tape started, the third monitor began. This was on the right, and lasted about 14 minutes. A single participant was asked to monitor all three video tapes at the same time.

The first experimental task involved a subject describing the instructional techniques or strategies and the management techniques or strategies observed being used in the classroom.

For the second task, participants viewed all three monitors again, this time with sound, and were asked to comment on what they heard and saw by typing a number designated by the experimenter to correspond to the given monitor they were watching.

The third task involved participants responding to nine questions concerning routines, content, motivation, learning environment, students' attitudes, teacher's expectations, teacher's roles, critical thinking skills, and the relationship that existed between the teacher and the students.

The fourth task was a measure of memory for specific details The participants responded "yes" or "no" to questions about what they had seen on each monitor.

Two researchers coded all the data independently, each using two coding systems. One coding system focused on the content of comments according to the following six categories: student behaviors; teacher behaviors; instructional techniques; management techniques; time use; and classroom environment. The second coding system focused on the nature of the comments and included five categories: description; interpretation; evaluation; conclusion; and suggestion.

The findings of the study suggested that experts were able to monitor, understand and interpret events in more detail and with more insight than either novices or advanced beginners. Experts, advanced beginners, and novices differed with respect to their ability to monitor and interpret simultaneous multiple events in the classroom. Expert teachers were not only more adept at monitoring all three screens simultaneously, but they were able to make sense of events that puzzled advanced beginners and baffled novices. Post hoc analyses revealed that novices made significantly fewer evaluative responses than advanced beginners and advanced beginners made significantly fewer evaluative comments than the experts. Experts, advanced beginners, and novices differed in their interpretations of what instructional strategies or techniques were used by the teacher. Advanced beginners recognized a method of presentation (lecture, etc.) They seemed to closely associate instruction with materials used. They were more preoccupied with the overhead projector than the novices, but less interested in the effect that instruction was having on student learning. Novices, like advanced beginners, seemed to focus on materials used. They were not able to identify instructional techniques (maybe because they were unfamiliar with the terminology).

Experts, advanced beginners, and novices differed in how they attended to the multidimensional nature of the classroom (emulated by the videos.) Experts, advanced beginners, and novices differed in their scanning patterns as they monitored multiple events in the classroom. Expert teachers scanned all three screens while subjects from the other two groups tended to focus their attention on the middle monitor. One explanation might be that the groups focused their attention on the teacher who appeared more frequently in the middle monitor than the other two. Experts, advanced beginners and novices differed in their ability to monitor classroom sounds and to use language to assist them in understanding interpreting and evaluating classroom events. Experts attended more to sound and to the instructional language used in the classroom than did members of the other two groups. They made specific comments about instructional content and classroom management. Experts, advanced beginners, and novices differed in terms of where they placed their primary focus when monitoring multiple events in a classroom. Novices focused and commented most on student behavior, especially behavior they felt to be inappropriate. Issues of managing student behavior and classroom control caught their attention and became the greatest source of their interest and concern.

Experts, advanced beginners, and novices performed similarly on tasks requiring judgement of content selection and on tasks requiring memory for nonmeaningful details. Experts, advanced beginners, and novices showed little difference in their perceptions of the appropriateness and accuracy of the lesson content (possibly due to difficulty interpreting the video). Experts, advanced beginners, and novices showed little difference in memory capacity for questions requiring specific memory for details. Experts, advanced beginners, and novices showed little difference in memory for salient or unusual events that they observed.

Thus, having expertise in a domain is not sufficient for effective teaching. Content pedagogical knowledge,

knowledge of how to present materials, must also be present.

Pedagogical knowledge, "knowledge about classroom management, organizations of classrooms, methods for motivating students, personal knowledge about students and families, and social-interactional skills," (Berliner, 1991, p. 147) plays a role in effective teaching, a role that cannot be played by content area knowledge alone. However, pedagogical knowledge, while traditionally studied as a heuristic, cannot be separated from the domain knowledge, as specific domains may require specific pedagogical approaches; in short, there may not be a pedagogical content strategy that can be applied globally (Berliner, 1991).

Experts' and Novices' Teaching Strategies

Teaching is a complex cognitive skill; researchers can attempt to recognize specific behaviors associated with the skill and the cognitive processes attributable to the behaviors that comprise the skill. Leinhardt (1986) equates expertise in teaching with student learning outcomes that are consistently above average. Learning is defined socially by the community and the school. Teaching expertise is always defined by at least one, and usually several, external criteria. That is, teaching expertise might be associated with student achievement, but also students' efficacy for a subject, motivation, or principal's evaluation.

Using an anthropological approach evolved since 1970, Leinhardt establishes a relationship with a teacher for at least two years so that the teacher is comfortable working with her. She then works with the teacher for three to 10 years. During the first year of data collection, two to four weeks of classroom observation are carried out. Sometimes audiotapes are made, sometimes field notes are taken. During the next phase, a block of lessons is observed and recorded in video tape. Before each lesson, the teacher is asked "What will happen?" and after each lesson, "How did it go?" The teacher views the videotape each day and discusses it. The discussion is recorded. The second and third years of study data are gathered on students. Students are given pre and post unit interviews, either in class or immediately afterward. Samples of students work are taken. All of the data are transcribed and indexed, annually. Leinhardt's methodology contrasts novices' and experts' performance in combination with an anthropological field-based approach. She attempts to combine concerns of subject matter with concerns of task. The strategy for much of her research is to examine one strand of teaching fairly carefully and then to go on to another until there are enough strands to weave together.

Leinhardt (1986) identifies several variables (routines, agendas, explanations, representations, and instructional dialogues) fundamental to teaching on which expert and novice teachers differed. Routines are conceptualized as automated activities that serve to reduce the cognitive processing load for both students and teacher. Examples of routines would be handing out papers or correcting the homework. Both students and teacher know exactly what behaviors are expected of them to the point where they perform without analyzing their actions.

Agendas are concise plans that describe the general set of goals and actions the teacher intends to implement within a 40-50 minute time frame. The agenda serves as a mental note pad for the teacher of what activities they are going to do in the classroom. Explanations represent the means by which new information is connected to prior knowledge. Explanations are situated within the semantics of the particular lesson. Good explanations constrain the meaning and application of new concepts.

Representations are analogies, the use of some type of physical representation of the targeted concept. They illustrate primary relationships, as in the use of flowing water to illustrate concepts of electricity. Although unarguably powerful teaching tools, representations are not without their liabilities. Representations may actually serve to generate misconceptions by oversimplifying features of the base and target in the interest of drawing the analogy. In addition, students must be able to make the leap from base to target, a process which may be beyond their cognitive capacity.

Instructional Dialogues are the formats through which problem posing, meaning clarification, shared responses and rationales, critiques, and consensus reaching are employed. Instructional dialogues are form, not content, oriented.

These variables function as threads that are woven together to create the fabric of teaching. They are inexorably linked in an actual teaching situation, but can be pulled apart for examination of how they contribute to the process of teaching, and how experts and novices may differ on these variables.

Patterns on which experts and novices differ begin to emerge when the strands are examined. Of course, in all research on teaching, there is a stated or tacit awareness of the necessity and dangers of partitioning the practice of teaching in order to study it. If the subject matter is the sole focus, features that in actual classroom activity would cripple a classroom are ignored. If the generic aspects of teaching are studied alone, then the substance and purpose of teaching are obscured. If management behaviors are studied alone, then the conditions for acquiring knowledge are excluded. If only the social context or political meaning is studied, then the task and rationale for teaching are lost.

Differences in the way experts and novices use routines becomes apparent upon analysis. Novices make less use of routines than experts. Lack of routines makes for more time spent in unproductive ways. The findings of Leinhardt's and Greeno's (1986) study of expert and novice teachers suggest that the routines of novice teachers lacked automaticity, resulting in greater instances of class disruption. That is, in the classes of expert teachers, both the teacher and the students had particular expectations for the execution of certain tasks, and these tasks were accomplished without interruption.

Experts and novices differed in their agendas as well (Leinhardt, 1986). The agendas of experts tended to include actions of teachers and students; refer to some prediction of student behavior; frequently suggest or make reference to a test or check that helps determine how to proceed; mention the location of the particular lesson in the wider spectrum of lessons; often include such overarching pedagogical rules as moving from the concrete to the abstract or such overarching content-driven rules as "this idea is useful in understanding the next idea." Expert teachers see the status of the students as changing throughout the course of the lesson. Novices show no sign of this capability.

Novices differ from experts in the quality of their explanations. Novices' explanations are often fragmented, a condition related to the lack of a cohesive agenda and the lack of routines.

Representations help to clarify why expert teachers usually present explanations that have a coherent quality to them, whereas novices give explanations that are island-like in terms of topics. Teachers who are not knowledgeable about a subject are not always able to analyze which features of a representation are good or not. It follows that teachers who are not knowledgeable about their students cannot construct representations that will be comprehensible to those students.

Teachers' Problem Solving

It has been suggested that experts and novices employ different strategies in their approaches to the multitude of problems that comprise the teaching task. Swanson, O'Connor, and Cooney (1990) examine the differences in strategy use between experts and novices in their approach to classroom management problems. Based on research suggesting that experts in most domains have a deeper understanding, more accurate representation of given problems, and thus, are able to more effectively focus on those variables most germane to the solution, ignoring superficial components, Swanson, O'Connor, and Cooney (1990) hypothesize that similar problem solving patterns will appear within the domain of teaching. Their subject pool, a group of 24 preservice and 24 mentor (expert) teachers, were assigned the task of solving some classroom discipline problems. The problems were described in six short vignettes, which teachers were to respond to by "thinking aloud." A further intent of the authors was to determine the degree to which an experimenter's directives upon which issues to focus impacted the experts' and novices' approaches to the problems; thus subjects were assigned either the "directive" condition, where they were told to focus on a) educational beliefs that influenced their decision, b) important information in the situation, c) additional information needed to formulate a decision, d) alternatives to their decision, and e) their final solution, or to the "nondirective" group, where they were not given such instruction.

A list of possible heuristics and subroutines was compiled apriori. These heuristics included a) the definition of the problem, b) the acquisition of data, and c) the interpretation of data. The heuristic subprocess definition of problem included mental processes such as listing given information, listing assumptions, listing possible questions, selecting evaluation criteria, assigning priorities, listing relevant and deleting irrelevant information, formulating hypotheses, defining predictions, and selecting questions. Acquisition of data included defining initial and goal states, identifying data needed, identifying algorithms needed, selecting and editing algorithms, executing the program, identifying feedback, and new information. The interpretation heuristic subprocess included matching data to predictions, determining truth values of predictions, extracting patterns from data, summarizing relevant patterns, and output conclusions.

A list of strategies was also identified. The general problem solver strategy included the defining of initial and goal state and identification of data needed. The feedback strategy included identification of feedback, tagging of new information, and organizing data. The pattern extraction strategy included the extraction of patterns from data, and the summarizing of relevant patterns. The hypothetico-deductive strategy included formulating hypotheses, defining predictions, matching data to predictions, and determining truth values of predictions. The evaluation strategy included listing of assumptions, selecting evaluative criteria, assigning priorities, listing relevant information, deleting irrelevant information, and editing algorithms. The basic problem solving strategy included listing given information, listing possible questions, selecting questions, identifying a set of available algorithms, selecting an algorithm, executing the program, and the output conclusions.

Essentially, the heuristics represent more global categorizations of mental processes while the strategies refer to more specific problem solving processes. Where heuristics typically include a presentation of the problem, a plan for acquiring information, and a means of interpreting and evaluating information, strategies describe specific problem solving processes (Swanson, O'Connor, & Cooney, 1990). Table 2 describes the general framework of the strategies.

Table 2

Explanation of Strategy Subroutines

Strategy Subroutine	Description
General Problem Solver	Assesses a subject's attempt to reach a goal state by taking a sequence of steps, each of which reduces the distance to the goal.
Feedback	Represents the subject's use of new information as it becomes available throughout the problem solving process.
Pattern Extraction	Represents an interpretation of information based on details in the problem situation.
Hypothetico-Deductive	Reflects if-then approaches through which predictions are affirmed or disconfirmed.
Evaluation	Represents a check on the adequacy of the hypothesis.
Basic Problem Solver	Keeps track of the directions the solver has

taken and utilizes
unexplored directions.

Note. From "An information processing analysis of expert and novice teachers' problem solving," by H. L. Swanson, J. E. O'Connor, & J. B. Cooney, 1990, American Educational Research Journal, 27, p. 533-556.

The think-aloud protocol data collected throughout the experiment was coded and then categorized into heuristic and strategy subroutines. Analysis of the data suggested that experts' responses contain more mental subroutines about classroom discipline than those of novices. Expert teachers used heuristic and strategy subroutines more frequently than novices, although under the directive condition, expert teachers' use of heuristics and strategies diminished. Novices tended to focus on surface details of the problem. Experts tended to emphasize to a much greater extent then novices, the Definition of Problem heuristic, which dealt with representing and defining the problem.

Novices tended to focus on problem solution, rather than a systematic testing of possible solutions. This suggests that novices' primary objective is to find a solution, rather than to define a problem and test possible solutions. The novice tends to represent problems in terms of possible solutions, instead of in terms of the problems.

These results parallel those in Greeno (1976) where novices focused on surface features of the geometry problem, seeking solutions rather than defining the problem, focusing on surface details because they did not have a deep enough understanding of the problem to differentiate between relevant and irrelevant or unimportant information. These findings may hold significant implications for other problems within the domain of teaching. For example, strategies which employ concentration on surface details when selecting educational software may result in teachers choosing packages that contain elaborate surface features, such as colorful graphics, but lack fundamental instructional principles. Teachers who have represented the problem of software selection as choosing a package that their students will find engaging may be more likely to focus on surface features than those who have represented the problem as choosing a package from which their students will achieve given learning objectives.

Summary

The expert novice paradigm provides a useful model for understanding the qualitative differences in the cognitive processes of expert and novice teachers, allowing researchers to focus on why and how novices make the transition to experts. The behavioral and cognitive patterns exhibited by expert and novice teachers mirror the those of experts and novices in other domains.

Like experts in other domains, expert teachers exhibit superior pattern recognition skills within their dominion. Expert teachers recognize patterns in student behavior, in curriculum materials, and in the structure and function of the classroom, which they interpret and respond to accordingly, just as solvers interpret features of problems and respond to these features in well-structured domains (Sabers, Cushing, & Berliner, 1991; Borko & Livingston, 1989.) For example, the computer technician observes the computer freeze every time the user opens more than one software package in Windows. The technician recognizes the pattern as one which occurs when there is insufficient memory in the computer and reacts accordingly. Likewise, the teacher recognizes that an open window onto the playground typically distracts some students, or that students do not complete homework if the teacher does not check it. Expert teachers use their pattern recognition skills in the same way experts from well-structured domains do. Novices, in contrast, do not have the pattern recognition skills that experts have. A novice teacher might not take note of the distracting open window until many of her students are disengaged from her lesson and fixated on the softball game.

Like experts in other domains, experts' well organized, comprehensive knowledge base allows them to access these patterns quickly, and to be able to distinguish what is important from what is not. For example, a skilled teacher might recognize that a careless error in multiplication is not a significant issue for a student who possesses deep understanding of the mathematical concepts.

Experts teachers also automate routine tasks associated with teaching, thus freeing up their cognitive capacities for more challenging circumstances (Leinhardt & Greeno, 1986). For example, an expert teacher might set up a system whereby students who complete their seatwork assignment automatically drop it into an in-box and know to take out the book they keep in their desk to read when they finish their work. The automaticity of this procedure prevents the teacher from having to attend to repeated student announcements that they have "finished" and need direction on what to do. Not bothered with this distraction, the teacher can focus his or her attention on, for example, a reading group, much the same way that an instrumentalist who automatically sight reads music can focus on his or her style or other aspects of making music without expending cognitive processing toward figuring out what each note is and where to place his or her fingers to produce the note.

Expert teachers also have the superior metacognitive skills that experts in other domains possess (Borko & Livingston, 1989). They can judge when their lessons are going well, and when they are not, and when they determine that the lessons are not going well, they can revise their strategies.

The problems typically encountered in the teaching process tend to be ill-structured, with a significant number of open constraints and a substantial number of possible approaches and viable solutions to the problem. For example, if the teacher wishes to teach a lesson on the solar system, there are a great many approaches this teacher can take, with a significant number of open constraints within the problem. The teacher must decide on the scope and depth of material to be presented, which may be determined in part by the curriculum and in part by the teacher's understanding of the students' knowledge base and abilities. The teacher must decide upon the format of the presentation: will there be a lecture, to what extent will the class be called upon to participate, what materials will be used to enhance the effectiveness of the presentation, such as videos or computer software, and what examples will be included in the presentation. The teacher must decide upon any activities that students will engage in to reinforce what is learned, and the teacher must decide upon a means of assessing what the students have learned.

One of the distinctions between teaching, and more well-structured domains, is its dynamic aspect. Each decision made in the interest of meeting the final goal affects the problem, sometimes changing it significantly, much in the way that plans made by that a person wishing to accomplish a given number of errands in a given amount of time may change each time the person makes a decision to stop a certain store. A decision to go down Main Street to the flower shop may be met with two hours of traffic, thus altering the plans considerably. Each decision has ramifications for later decisions in a way that cannot always be foreseen.

A teacher's decision to show a video on the solar system opens up a realm of other considerations: will the students attend to the video, or will they fall asleep or act out? How will the teacher handle the students who are not engaged in the video?

Well-structured problems do not present the solver with as many contingencies. While more than one strategy may be effective in the solving of the problem, the number of contingencies resulting from the execution of a strategy is fairly limited. That is, the dividend and divisor will remain constant no matter which strategy is taken to solve the long division problem; some strategies will be less effective in terms of expedience, but no strategy will cause the divisor to suddenly become another quantity. This predictability is, to an extent, present in many ill-structured problems. For example, in the problem of arranging furniture in a house, the variables have fairly predictable properties; putting the chair too near the door might result in people opening the door into it, (not a happy prospect for the chair) but the chair will not move itself to another corner of the room or incite the other pieces of furniture to riot. The error of incorrect chair placement does not have extensive ramifications.

Because of the large number of open constraints involved in problems associated with human response, the actions or decisions of teachers can result in unpredictable consequences that reach beyond the scope of a student not learning a particular lesson. A poorly presented lesson may result in having some students lose all interest in the subject. A lack of classroom organization can result in a chaotic atmosphere that negatively impacts all attempts at instruction. Individual actions or decisions can have a significant effect on problem state.

It can be argued that expertise in teaching is related to the ability to make accurate predictions about the behaviors and cognitive processes of students. That is, if the goal of teaching is to teach, or, to modify the schemas that students have about a given topic, teachers must make predictions about what those schema are, and about the most effective way to impart the information to those particular students. When teachers plan a lesson, they are trying to make predictions about the interaction of material they wish to cover, the classroom environment, and the attributes of their students. It can be argued that when teachers make decisions about which software packages to select, they are doing this as well, since an educational software package is another form of presenting a lesson or classroom activity. Expert teachers will presumably use the same skills they use to evaluate these electronic lessons and activities that they use in evaluating the traditional lessons. They would be anticipating some of the same problems and focusing on some of the same variables that they would when planning other lessons.

Viewing Software Selection as an Ill-Structured Problem

Software selection can be viewed as an ill-structured problem with the solution to the problem being the identification of a software package that meets the teacher's learning objectives for his or her students. Like teaching and other ill-structured problems, the number of open constraints is considerable. There are a large number of possible screen designs, for example, that would be considered by the problem solving community to be effective in terms of clarity. There are a multitude of examples that could be deemed appropriate as exemplars for any given lesson. As with other ill-structured problems, a large number of solutions will satisfy.

Reitman (1965) suggests that the source of ambiguity is related to the open constraint set; when a large number of values can be assigned to the open constraint set, the number of potential solutions will be relatively large. For example, a blue background with yellow text may satisfy the constraint of eye appeal for one problem solver but a white background with blue text may also function as an appropriate value for the constraint. The solution to the problem of whether or not to select the software may be binary but the routes to the solution and the solution itself are not. The problem of whether or not to select a given package for use in the classroom may be represented as whether or not a given package is viewed by the teacher as being instructionally effective. To an extent, instructional effectiveness is a subjective construct; there are a variety of measures that might be used to ascertain the effectiveness of a software package, and a variety of variables associated with effective instruction.

The following chapter will examine the variables associated with educational software and discuss how these variables contribute to the instructional effectiveness of the software.

CHAPTER III

Software Evaluation as an Ill-Structured Problem

Selecting software involves making a decision, much in the way as the Johnson (1988) study of admissions officers selecting candidates for admission to medical school involved making a decision. The decision making process requires the identification of a set of criteria or variables seen as significant contributing factors in the desired outcome, the weighting of these variables for their significance in the final decision, and the evaluation of their state in terms of whether they support or oppose the particular desired outcome. For example, a teacher deciding on the purchase of a software package must 1) identify which variables will be of importance, for example, accuracy of content, 2) determine how well that the package represents that variable, such as the content was or was not very accurate, and finally, 3) decide if that is even of any consequence. For example, maybe it really does not matter if the package has animation.

As with other problem solving tasks, pattern recognition skill is a factor that will contribute to the skill of the decision maker. That is, experts are more likely to quickly identify the variables of interest, despite the fact that they may not be overt or obvious. For example, expert teachers might quickly note that a software package provides no feedback other than "correct" or "incorrect" and determine this to be a weakness, based on what they know of learning theory.

Practical Approaches to Software Evaluation

A variety of software evaluation strategies are available to teachers. McDougal and Squires (1995) distinguish between assessments made by the teacher prior to actually using the software with his or her class, termed selection; assessments for which the goal is a review of the software including descriptions of its features and attributes (usually for the purpose of assessing a software package for a larger audience) termed review; and formative and summative evaluation. The former refers to the assessment of the package during the development stage, presumably largely the province of the instructional design team. The latter refers to evaluations made after the package has been used, with the purpose of the evaluation being to determine how well the achievement aims of the software have been met.

There are strengths and weaknesses in each of these assessments. The selection option requires teachers to make predictions about how their students will interact with the software. Thus, they never actually have data substantiating its effectiveness when they make their decision to use it. If teachers are very knowledgeable about their students, they may be able to make sound decisions without the benefit of a trial run. However, if they are not, they may not be making informed decisions. An advantage to the <u>selection</u> option is apparent when considering that since measures of achievement may vary somewhat from teacher to teacher depending on the given goal they had for the use of the software, teachers are not locked into using or not using a package based on some criterion that may not be relevant to what their own objectives for the software might be. For example, a teacher might want to have his or her students use a package strictly to get them motivated to learn more about the topic; thus, it would be the motivation that would be measured, not their scores on a content-based assessment.

The <u>review</u> assessment has the weakness of being subjective. If a small number of teachers are reviewing the package, the subjectivity may be fairly high. However, any review still provides information, which is, in some cases, better than no information. If the review includes less subjective information like length of modules, presence or absence of on-line help, this might increase the usefulness of the review.

The <u>formative</u> evaluation has the strength of allowing teachers to participate in the development process; however, this is not always an option for the teachers who will eventually use the package. Teachers without technical or programming experience also might not have a clear sense of what is feasible in terms of the limitations of the programming language, of what can be done, for what cost. However, by the same token, programmers do not always have a strong sense of what is instructionally effective.

Finally, the summative evaluation may provide the clearest picture of the effectiveness of the software, but it does require many of the considerations of any research study. Data must be collected, analyzed, and the subject pool should be generalizable to other students for the study to be of merit. A willingness to participate and an availability of time and resources is necessary to effect this type of assessment.

Certain variables will still be significant, regardless of the means of assessment used. In situations where the selection form of assessment is used, teachers make predictions about the effectiveness of the software by focusing on these variables.

Structural Framework of Software

Different presentations are often appropriate for different learning situations. For example, a long talk and chalk lecture might not be appropriate for young school aged children, but a lesson on the solar system that involves creating plaster-of-paris mobiles of the planets might not be appropriate for high school students. One of the key variables to consider in software evaluation is the general design of the software. As different presentations may be appropriate for different lessons and different audiences, the same is true for software.

Instructional software can be conceived of in terms of models that function to allow greater understanding of the developer's objective for the software and how the software can be most effectively used. They also function to assist software designers and developers in creating effective software.

Edmonds, Branch, and Mukherjee (1994) developed a matrix of instructional design models which include 15 existing models, and which can be used for purposes of comparing these models and identifying key aspects of future models. The matrix is comprised of six variables on which instructional design models can be categorized. The variables are level, context, expertise level, knowledge structure, and orientation.

The level feature of a software package is used to describe the particular learning objective or eventual application of the software. Categories of levels include module, lesson, unit, course, curriculum, institutional, and mass.

The module level focuses on the smallest unit of instruction, typically the micro details of a lesson, like learning a given formula. The lesson level covers the material that would occur in several episodes of instruction; the unit level determines specific topics to be learned.

Course level material covers the information presented in a course, such as introductory algebra; curriculum level extends to the curriculum that might be included in a Bachelor's degree program in mathematics at a certain college. Institutional level instructional design would be appropriate for an institution such as the U.S. Army, and mass level typically encompasses instruction that would be nationwide or global.

The context refers to where the software is to be used. The matrix lists four possibilities: K-12, higher education, business, and government. K-12 is elementary through high school; higher education refers to colleges, and the business and government categories are self-explanatory.

The categories of level and context are most easily identifiable for the software user. Certainly an instructor will have an idea if the software he or she is evaluating would be more appropriate for a 10th grade math class than a graduate class, and it is also usually easy to differentiate between a software package that is intended to cover an entire course, and one that is intended to cover a specific topic. The importance of making these distinctions extends to the effectiveness of the software. A student who needs help with long division may not be helped by a package that is devoted to the entire third grade math curriculum, not only because of the amount of material likely to be covered, but because of the treatment afforded the given topics (although of course there are likely to be course-related packages that do an excellent job with individual topics.)

The orientation of the software refers to the theoretical instructional framework. The orientation can be prescriptive or descriptive, or both. Descriptive models

> "describe a given learning environment and speculate how the variables of interest will be affected in such an environment. Prescriptive models outline how a learning environment can be altered or constructed in order to affect the variables of interest in a certain way or bring out the desired outcome...Instructional models which are both descriptive and prescriptive speculate and recommend organizational strategies, and are typically comprehensive and explicit in the amount of detail

for application" (Edmonds et al, 1994, p. 60).

It could be argued that prescriptive models tend to be developed around or have more of a formula for what should be included to achieve some specific result. For example, users will do exercise A and event A will occur on the screen and users will respond to event A in a specific way to enact event B. In short, there is a prescription to be followed to achieve a given outcome, and this prescription involves the manipulation of specific variables in the learning environment.

In contrast, the descriptive model is much less structured. There is no prescription for what must take place in order to effect any specific outcome. Rather, the designer is instead speculating, based on experience in instruction and learning, that the interaction with the learning environment may lead to certain outcomes. The combination model incorporates both of these perspectives (Edmonds et al, 1994.)

Knowledge structure determines the type of task the user is intended to perform. A declarative model is based on a learner performing specific tasks with the goal of meeting a criterion. Declarative models generally favor lower-ability learners, are often drill-and-practice exercises, and may reduce the learning efficiency of higher-ability learners. A procedural model is discovery oriented, and incorporates norm-referenced evaluation. Such models best serve higher-ability learners as they require the user to be comfortable abstracting knowledge from the learning environment (Edmonds et al, 1994).

Instructional design models also vary according to the design experience and ability of the designer. Some models provide step-by-step instruction for the designer, algorithms they can use in the construction of their designs; others rely on the designer's intuition or experience as the guide, providing very broad heuristics for the designer. The level of expertise category of the matrix is divided into expert, intermediate, and novice levels (Edmonds et al, 1994).

Finally, the matrix includes a structure category. Structure refers to the underlying theoretical origin of the instructional design. The dominant approach in most instructional software is based on system theory (Edmonds et al, 1994). A system is a "...regularly interacting or interdependent group of items forming a unified whole...Systems are characterized by input data and output data processes that are clearly specified; an explicitly defined locus of control; and interrelationships of various components which comprise the system" (Edmonds et al, 1994, p. 56).

Essentially, the outcomes of each component directly or indirectly impact every other component in the system.

Hard systems designs emphasize identifying problems and seeking solutions to these problems.

Alternatives to systems-based design include soft system designs and intuitive designs. These alternative designs are not constructed around the premise that there is a problem for which a solution should be found and applied, but rather around the premise that there is a situation which may be accommodated by a certain action. It can be argued that this alternative paradigm is not conceptually unlike the alternatives to process-product instruction, in that the designer begins to move away from the strict, criterion-based assessments which assess whether or not a specific learning goal has been achieved. Like the required expertise parameter, the structure parameter is seen as a continuum ranging from hard systems to intuitive designs, with intuitive designs falling farthest from the systems paradigm.

An understanding of these six variables and their impact on a software package may prove helpful in software selection decisions. Such knowledge allows instructors to make more informed decisions about how well suited the program is to their students' needs and their own learning objectives.

Variables Identified as Consequential in Software Evaluation

The effectiveness of instructional software is ultimately evaluated by the degree to which learning outcomes are achieved (Castellan, 1993). Outcomes may be criterion-referenced (Duncan, 1993) or norm-referenced. Norm-referenced assessments measure performance based on a comparison of other scores in a given distribution. Criterion-referenced outcomes (Baker & O'Neil in Duncan, 1993), more commonly employed, are assessed through performance on a well-defined behavior domain, such as a multiple choice test. It requires the generation of a set of test items from the given knowledge domain. A large number of variables mediate the degree of learning which takes place through use of software. Many of these variables are external to the software, how teachers use the software, the classroom atmosphere, and affective and cognitive student characteristics, will impact on the effectiveness of the package. It is the interaction of these external variables with the variables present in the software that will make for effective instruction.

The main purpose of instructional software is to teach. When evaluating the potential effectiveness of a software package, it is useful to consider the role in the software package of key variables that operate during teaching and learning situations (Shuell & Schueckler, 1989). Cohen, Chechile, Smith, Tsai, & Burns, (1994) examined a software package based on the generative approach to learning using statistics as the domain. They analyzed how students interacted with ConStatS, a program developed at Tufts University for teaching introductory probability and statistics. Their analysis was inspired by three goals. The first was to evaluate the effectiveness of the software; the second was to assess ways in which to increase the effectiveness of the software; and the final goal was to ascertain whether or not students were engaged in the types of activities that were associated with increased understanding.

A method of analysis known as Trace, which allows researchers to determine the time spent and activities engaged in for given parts of the software, was developed and employed. The Trace program runs in the background of the software to be analyzed and records the time individuals spend on given screens and what interactions are performed. Researchers then have data that can assist them in determining patterns of user behavior, and how such patterns correlate with learning outcomes.

Cohen et al (1995) developed a model of the way students interact with a software package. The model consists of four types of interactions that can take place during use of the program. The first type of interaction is scaffolding. Scaffolding occurs when the student uses elements of the software, for example, text, to develop an understanding of a given concept. They serve as "cognitive preparation" for the student who will then be asked to use the concept. One could argue that scaffolding interactions are analogous to a professor's lecturing; the student is being exposed to information that he or she will eventually use.

The second type of interaction defined is investigation. Investigations are interactions that allow the student to manipulate the concept. These are typically experiments and simulations, and can be animations. Investigations do not require active behavioral interactivity; rather they provide a context through which students can construct a new understanding of a concept.

It can be argued that the distinction between scaffolding and investigations are measured by the result--both appear to provide the student with information--but it is through investigations that the student develops the deeper conceptual understanding that typifies learning. However, if interactions need not be interactive, the distinction appears to become one of text versus image, with the scaffolding (text) setting the student up for the video to follow. The assumption being made is that the video presentation will be more effective than the text, a factor which is not necessarily true.

The third interaction is reformulation, defined as the cognitive change that results immediately following an investigation and that continues until the next interaction. Presumably this can be assessed via some assessment in the software. In addition, subsequent interactions were examined providing insight into the temporal progress of the student's understanding.

The final interaction is navigation, the interactions that take the student from one part of the program to another.

The model takes the premise that it is the investigation that leads to the cognitive resolution, or that learning is a generative process, with students discovering the concept via their experiences. The Trace program is able to provide evidence supporting or rejecting this assumption by providing researchers with information about how students are actually using the software (Cohen et al, 1994; Cohen, Tsai, & Chechile, 1995). Presumably, those who spend little time investigating would do less well than those who devote greater energies to this type of interaction.

The study began in 1992 and ran for three years. College students from three universities participated. Seven classes with 327 students participated as experimental groups, and three classes with a total of 63 students participated as control groups. Two of the control group classes were taught by a member of the team that designed ConStatS. Control groups used tool based statistics software for mainframe and personal computers but did not use ConStatS. Two-hundred and twenty-six experimental subjects were enrolled in an introductory statistics course for psychology at Tufts University. Other subjects were enrolled in statistics courses for biology, education, child study, and economics. Presumably, individual differences between the way the instructors of these courses used ConStatS exist. A brief survey was given to students to determine if they felt the software was useful and used effectively by the instructor, and comments and suggestions by students, instructors, and teaching assistants were always welcome. Researchers also used a 7-point survey of course content to assess which topics in the software were emphasized in lectures, homework, and texts.

The outcome assessment instrument consisted of short answer questions to test conceptual understanding. A collection of questions related to the content of the software was generated and from this large sample of approximately 1000 questions, 103 questions which best represented the key conceptual issues were selected for inclusion into three tests covering similar, although not identical, content. The questions typically required students to transfer knowledge into novel situations, distinguishing between deep conceptual understanding and declarative or procedural knowledge. In addition, a ten-question math test was given to ensure students had adequate skills for using the program.

Questions on the comprehension test were scored on a 5-point scale with 1 representing no understanding, 2 - little understanding, 3 - moderate understanding, 4 - good understanding, and 5 - excellent understanding. Results of this test indicated that the experimental group outperformed the control group with the experimental group outperforming the control group on 92 of the 103 questions.

Analyses of the results focused on the specific interactions that correlated with increased learning, and on the general patterns of interactions. It was hypothesized that certain interactions would be present if a student was learning by active experimentation, and that these interactions indicated that the student had a well-formulated question in mind.

Results of the analyses suggest that these hypotheses can be substantiated. To achieve an accurate answer to a question about discrete probability distributions, for example, students needed to perform a few experiments (those who performed three consecutive experiments gave correct answers 46% of the time compared with students who did not and gave correct answers only 10% of the time.) In addition, students performing the experiments averaged about twice as many total interactions as did other students over the same period of time. Other examples further support the researchers' hypotheses. Students who manipulated the distribution sample by increasing sample size and then examining the mean performed better on a question about the relationship between sample size and true mean than those who did not. Students performing the interaction averaged 80 percent on the question; those who did not, averaged 60 percent. (The article does not indicate if this is statistically significant.)

These results suggest that interactions that allow students to manipulate variables and to observe outcomes appear to be positively correlated with learning gains. Given this assumption, it follows that one of the attributes of effective software would be the degree to which it encourages or perhaps requires students to perform these interactions (assuming the interactions are instructionally sound from the perspective of accuracy and ease of understanding). If the software cannot be previewed with students, teachers must be able to predict the degree to which it encourages interactions.

Indeed, when software must be selected for classroom use prior to its being evaluated with students in the classroom, teachers must then select variables that will serve as predictors for effectiveness. Castellan (1993) identified a set of criteria that can be used for evaluating instructional software that does not require that a student first use it and then be tested on the information he or she should have learned as a result of this exposure.

The evaluation methodology termed strategic evaluation, includes examination of the software's technical accuracy, pedagogical soundness, substantive fidelity, integrative flexibility, and cyclic improvement. Castellan argues that "the determination of whether the technology works is in fact evaluation against a criterion that, although it may not be well articulated, nonetheless can be judged" (p. 234). In short, Castellan aims to assemble a collection of variables that can be associated with software that "works."

Technical accuracy refers to how well the software actually functions from a technical perspective. Inherent in software is the technical aspect; if the programming is faulty, the most instructionally sound program is quickly abandoned. In addition, technical configurations may vary from network to network, school to school, computer to computer, and the status of the software's ability to run on a variety of platforms and configurations become important.

Pedagogical soundness describes software that is sensitive to the "fundamental behavioral and cognitive processes involved in learning, thinking, and problem solving" (p. 234). Castellan postulates several factors that fall under the dominion of pedagogy. These factors include instructional goals, function of technology in the lesson, flexibility of the technology, a means of self-assessment, and transfer of concepts presented through the technology.

Instructional goals should be clearly articulated for the student. Castellan asserts that most students do not discover goals not explicitly stated for them. Students who know the goals can monitor their progress, a contention not limited to the realm of instructional software.

Students should be clear about the role that the technology is playing in the presentation of the lesson. That is, students should understand the differences between simulations and actual events, and should not get so caught up in the technology itself as to lose sight of the point of any given demonstration. On a related note, technology should not be used simply for the sake of using technology. If a given concept is not suited to presentation via software, it should not be presented via software. Likewise, the technology should be used at appropriate times throughout the course; a new package need not be used immediately if the topic it is covering will not be brought up in class for three weeks. Again, technology should not blind users to the fact that the issue is instruction, not special effects.

The technology should be flexible enough to allow students to get an overview of the material, to get detailed information, or to review specific material. Students should not have to go through material they have already seen and need no help with, in order to get to the material they really need to review.

Castellan's recommendation that software encourage exploration, testing, and concept acquisition mirrors recommendations for other learning modalities. Having students explore material often provides an effective way for them to integrate into their existing schemas new information. Cohen et al (1995) found that students who experimented with concepts performed better on learning outcome assessments.

A means of self-assessment should be included to help students monitor their progress. Finally, as in all forms of instruction, the student should be able to transfer what he or she has learned to other contexts. Unlike the other suggestions, this final recommendation cannot actually be assessed until the student uses the software. However, some form of explicit mapping from the analogy to the base may aid in the transfer. For example, if a game is used to explain a concept, some concrete references to how the actual concept relates to the game should exist.

Substantative fidelity refers to the accuracy of the information presented in the software. The content of the software must be accurate; faulty information is no more than that, regardless of the sophistication of the technology. Castellan also poses the question of whether or not the material is worth learning. That is, he is concerned the material has some degree of importance to the overall learning objectives.

Technical flexibility refers to how easily the software can be integrated into the curriculum and the ease with which it can be modified to better fit the instructor's needs. The usefulness of a package will be diminished if it can only be used in a very limited capacity under very specific conditions.

Castellan suggests that attention to these criteria can aid instructors in selecting effective software. When the actual evaluations take place cannot be discounted. Evaluations should be made fairly regularly to allow for cyclic improvement; that is, software should be evaluated before, during, and after a course.

While outcome assessment is often associated with achievement measures, other variables such as student attitudes towards the software, their perceptions of its usefulness, and the degree to which its use stimulates interest in course content also provide valuable information for evaluators (Duncan, 1993). Research into effective teaching suggests that there are six functions performed by effective teachers: These are

> "1) daily review and checking of previous work--including reteaching when necessary; 2) presenting the material to be learned--including a short statement of objectives or goals or a structured overview; 3) guided practice--including checks for student understanding of the material to be learned; 4) providing correctives and feedback to the learner that corresponds to the type of response that was made; 5) independent practice appropriate for the material being learned; 6) systematic review of previously learned material." (Shuell & Schueckler, 1989, p. 138).

In addition, learning tends to be most effective when teachers have a goal, and students are aware of the learning objective (Shuell & Schueckler, 1989). Students' prior knowledge has been shown to be positively correlated with active learning, that is, learning situations in which students are actively engaged, and situated instruction, that is, learning circumstances where students are able to see realistic, true-to-life applications of the concept (Brown, Collins, & Duguid, 1989). While the instructional situation may vary, for example, content will vary, nature of the material (declarative information such as knowledge of the capitals of states, or conceptual information such as the historical significance of the passing of a certain bill) will vary, the particular instructional variables employed vary only in the way they are applied for given situations (Shuell & Schueckler, 1989). Feedback, as an example, is appropriate in a variety of instructional situations, although the nature of the feedback may vary. A simple "correct" or "incorrect" may be appropriate for practicing multiplication tables, but much more extensive feedback might be required for the answer to the question "What is the theme of a novel.

It should also be noted that an instructional variable may take on different incarnations, that is, ascertaining prior knowledge of a subject may take the form of an oral survey, a short written test, or a general class discussion (Shuell & Schueckler, 1989). This is particularly relevant when evaluating software packages on the presence of absence or certain instructional variables; motivation may appear in a dramatically or slightly different form from the way it appears in a traditional learning situation.

Shuell & Schueckler (1989) developed a nineteen item software evaluation form to help evaluators determine the degree to which software was based on instructional principles. The instrument was derived from research on teaching and learning, reviewed by advanced doctoral students, and revised. Each item has a six-point scale on which evaluators rate software on given criteria largely described in the six behaviors associated with effective teaching.

Software is rated from "Low" (1-3) on the scale, to "High" (4-6) on the scale. Items assess whether objectives are stated in terms of what the students will do, whether prior knowledge is assessed, whether prerequisite knowledge is reviewed, and if the program provides for reteaching. Also included are items assessing how appropriately the material is presented, whether the program includes appropriate feedback, and whether it assesses the degree to which the student is learning. Finally, the items measure the degree to which guided practice and independent practice are included, how motivating the program is, the degree to which graphics are used, whether or not the program provides for closure, and the degree to which the program has provisions for profiles of students' progress, to be reviewed by the teacher.

Sixteen software packages were randomly selected from a pool of approximate 1,150 programs from the library maintained by the Software Evaluation Project of the State University of New York at Buffalo, representing four major content areas; math, social studies, language arts, and science. A variety of grade levels and program types (drill and practice, simulation, etc.) were represented.

The software was evaluated by both Shuell and Schueckler. At the time of the publication of the article describing the study, Dr. Thomas J. Shuell worked in the Department of Counseling and Educational Psychology at the State University of New York, College at Buffalo. The evaluators discussed any disagreement on ratings and if differences could not be resolved, the average score of the two evaluators was recorded.

The overall mean for all of the software evaluated on all of the criteria was 2.92. Variables on which programs were more highly rated included:

1) presentation of the to-be-learned material in steps or blocks in age, grade, or subject-appropriate manner; 2) consistency between the program's stated objectives and the actual program; 3) examples; and 4) opportunities for independent practice.

Variables on which programs earned lower ratings included:

1) degree to which the students are informed of goals or objectives of the program in an accessible

manner; 2) assessment of prerequisite knowledge for learning the material presented; 3) review of prerequisite knowledge; 4) reteaching of information student is lacking; 5) providing anticipatory set (that is, an introduction, or advance organizer, or similar cognitive assistance); and 6) a means of closure. Differences across curriculum areas and program types could be attributed to small sample size and were not focused on.

In sum, the variables of interest were presence of clear objectives, attentiveness to prerequisite knowledge, presentation of material, guided practice, reteaching and independent practice, motivation and graphics, and recordkeeping. The presentation of clear objectives is essentially the goal-stating operation. The software evaluated in the study received low ratings in this area.

The results of the study suggest poor performance in the area of prerequisite knowledge assessment. A weakness in this variable may prove significant as learning theory suggests that the most effective instruction takes place when students have been asked to recall prior knowledge, and have the requisite knowledge on which to build the new knowledge. Presumably, teachers can fill in for software packages in these areas.

The sample of software packages evaluated performed relatively high on presentation of material, the variable that includes presenting material in appropriate blocks, the software being consistent with the stated objectives, and the inclusion of appropriate examples.

Guided practice, the opportunity for students to practice a new skill with accompanied support, was lacking in the sample, although reading and language arts programs received higher ratings in this area than the other programs. The software did not perform an effective job of assessing student progress.

The software sample received low ratings on reteaching but high ratings on independent practice, presumably because drill and practice programs essentially revolve around independent practice. It is probable that given the dynamic nature of most packages, software developers assume that students who need to review material will look again at those parts of the software relevant to their needs. However, to increase the chances of this actually occurring, the software might have to provide a means of assessment of learning that the student could access, and perhaps some prescriptive instruction on places in the package the student should repeat. However, the possibility still exists that the reason the student didn't learn the material the first time is that it was presented in a manner ill-suited to the student's own learning style. Where a teacher may be able to offer a variety of presentations in the reteaching segment of his or her lesson, the program cannot spontaneously structure a new presentation based on student cues.

The sample packages did not score high on motivation or graphics, but again, a sample of 16 may not be large enough to ensure reliability, particularly given the population of over 1000 software packages. In addition, these packages were reviewed by only two individuals. It is interesting to note, however, that even the most elaborate graphics are subject to the capabilities of the computer; that is, the most beautiful colors will appear in black and white on a monochrome monitor. This factor is important for evaluators to consider; a software package being selected for its remarkable graphics may not be worth purchasing for an old system.

Record keeping received low ratings; it is suggested that perhaps software developers are not certain as to the kind of information teachers wish to have recorded, for the programming required to keep records is well within the range of the technology.

The Educational Products Information Exchange (EPIE) has identified variables that instructors should be attentive to in making software selection decisions. The variables describe user characteristics, classroom characteristics, instructional goal characteristics, and hardware and software characteristics.

Hardware characteristics refer to the capabilities of the computer and any accompanying peripherals. If a computer is lacking in memory requirements for a software package, the package is likely to run poorly. Lack of hard disk space may make it impossible to install. On a related note, the some packages require specific software to run, such as Microsoft Windows, or certain versions of DOS. Instructors who purchase a package for which their computers do not meet hardware requirements may be faced with either upgrading their systems, an option which may be prohibitively expensive, or returning the software.

The importance of attentiveness to user characteristics is of particular importance as software packages, like other instructional media, are typically developed for a specific audience, and their effectiveness may be diminished if used with an audience for which they were not intended. EPIE recommends considering users' age or grade range, and whether or not they meet the competencies suggested by the software developer. For example, a user who does not yet read may not benefit from a package which requires him or her to read instructions in order to use it.

Software characteristics include components of the software; that is, number of disks, documentation, teacher support materials; pricing; curriculum role; content; and pedagogical approach. Documentation and teacher manuals may impact on how effectively the software can be used. These too can be evaluated on their clarity, accuracy, and effectiveness.

EPIE describes curriculum role as the domain or subject that the software addresses, how in-depth the treatment of the material is, that is, supplemental or comprehensive for example, and the approach the software is taking, either drill and practice, educational gaming, simulation, logic/problem solving or an evaluator-defined category. The EPIE evaluation form suggests domains may fall under the categories of mathematics, spelling, language arts, reading, social studies, science, or some other evaluator-defined domain. Instructors can match their needs of how sophisticated the coverage of the topic is with an appropriate software package and a desired approach.

Drill and practice programs typically function to review declarative knowledge with students, allowing them to practice to an established criterion. Flash cards serve as a drill and practice exercise; analogous exercises in software packages are termed drill and practice. Tutorials are essentially electronic tutors; they are packages which present information, and usually allow students to participate in the lesson depending on the degree of interactivity involved. Educational gaming packages include games whose fundamental goal is to provide some instructional benefit. Simulation programs depict some phenomenon by using the properties of the computer that make possible animation, video, audio, and other multimedia. They allow a student to watch an event as if the event were actually taking place. Logic/problem solving programs endeavor to teach students to solve problems or perform cognitive functions more sophisticated than the rote memorization associated with drill and practice (Bitter, 1993; Roblyer, Edwards, & Havriluk, 1997).

These categories are not mutually exclusive; indeed many packages incorporate aspects of most or all. It can be inferred that instructors may select from these different approaches depending on the topic they are covering. For example, the memorization of the multiplication tables may be enhanced through a drill and practice exercise, but conceptual mathematical thinking, such as the solving of complex word problems might be best covered by a problem-solving oriented package. If frogs are not available for dissection, students can still have that learning experience through simulation programs. A student who wishes a step-by-step approach to learning to calculate the standard deviation might be well served by a tutorial. Finally, educational games are often based on one or more of the above approaches but offer additional motivation inherent in game playing (Roblyer, Edwards, & Havriluk, 1997).

Issues such as appropriateness for users in terms of ability, appropriateness of scope of material, readability in terms of vocabulary and sentence structure, clarity of presentation, accuracy, and freedom from stereotyping fall under the category of content.

Also included by EPIE under the heading of content is graphics, specifically their presence, their function, and their appeal. On a related issue, audio is considered, with much the same focus questions, essentially how the audio impacts on the program. Finally, EPIE suggests support materials should be examined and their role in the software package identified.

EPIE's category of methods and approach used in the software includes such variables as technical quality, documentation, user control, feedback, and random generation. Technical quality encompasses the ease with which the software can be used, the consistency with which the program uses given keystrokes or actions, the degree to which it is free of programming errors, and the terms of the warrantee. Documentation refers to any teachers' guides or technical explanations included with the software. User control refers to the degree to which users can direct their course of action when using the software.

Feedback describes the software's response to student interaction with it. According to the EPIE evaluation sheet, evaluators should be attentive to the appropriateness of the feedback, and its function in the software. Some feedback serves to remediate; feedback may also be reinforcing, or threatening. Sometimes feedback may unintentionally reinforce behavior that is not desirable, as, for example, in the case where an appealing graphic or humorous sound occurs after an incorrect answer.

Random generation is a component of the actual programming. Computers have the capability of generating responses at random. This feature can add authenticity to simulations or games. For example, instructional games often feature events that occur by chance, lending credibility to their scenarios.

Finally, EPIE recommends consideration of the evaluation component, that is, the way in which the program provides for the measuring of goals and objectives.

An overview of evaluation forms, which typically provide a more systematic and structured approach to evaluation, and software reviews which tend to provide a more subjective appraisal (Schueckler & Shuell, 1989), yielded a collection of criteria or variables upon which software is typically judged. Twenty-six criteria were identified by (Schueckler & Shuell, 1989) in nineteen evaluation forms and software reviews. The criterion were then clustered into four major categories, fundamental program characteristics, instructional concerns, principles of learning and teaching, and overall ratings.

The fundamental program characteristics are then subdivided as follows: basic information, which includes program name, subject area, publisher, and cost of the package; technical aspects which include hardware, that is, specification of computer make, model, memory capacity, and number of disk drives necessary for running the program; and additional hardware which refers to all other peripherals; type of program which refers to whether the program can be classified as an authoring system, drill and practice package, game, problem solving package, simulation, word processor, utility, or some combination; operational concerns which refer to how user-friendly the program is, how smoothly it runs with respect to lack of "bugs," and the visual elements of the program, e.g., sound, graphics, menus, in short, the user interface; directions for use, which refers to any on-screen prompts or off screen documentation; and finally, execution time, or the time required to load and use the program.

Instructional concerns are subdivided into the following categories: Social interaction includes the learning environment of the program, that is, a competitive or cooperative atmosphere, and the size of the group that it is designed for; user orientation which refers to where the control of the software lies--it can be teacher oriented, meaning the teacher can manipulate the level of difficulty, content, speed or presentation, or that teacher intervention is required for effective use, or the program can be student oriented, meaning the student is essentially the control figure with respect to level of difficulty, or other manipulations; prerequisite skills stated refers to whether the program states the prior knowledge required for use; educational objectives stated refers to whether the program states educational objectives or goals, educational content refers to the accuracy and educational value of the content, material presentation describes whether the material is presented in appropriate increments, and contains some means of assessing students' comprehension, and appropriate use of computer assesses whether the features of the computer are well-utilized.

Principles of learning and teaching is subdivided into the following categories: motivation describes whether or not the program is stimulating and interesting, for example, is it interactive enough;

feedback assesses the appropriateness of the program's responses to student input; record/score keeping refers to the program's recording, for immediate access, of the accuracy of students' responses; cognitive level determined describes whether the software describes the type of cognitive ability required, that is, the nature of the students' tasks--knowledge, application, evaluation, etc.; and evaluative teaching methods used which refers to the assessment of the students' work in terms of scores on a diagnostic test or some similar measure.

Finally, the overall rating category may consist of a compilation of scores or a subjective rating.

Instrumentation Used In Software Evaluation

One way to formulate a search for variables that are associated with effective software is to examine software evaluation instruments as was done by Schueckler & Shuell, (1989), and described above. Examination of these instruments yielded a set of dimensions on which evaluators may focus when making selection judgments.

However, while many formal software evaluation instruments exist, they are not without weaknesses. Understanding where the weaknesses lie is useful for evaluators, not only in terms of better realizing the reliability of the instrument, should the evaluator which to use it, but the role that the different variables and dimensions can have when viewed from different perspectives. For example, "ease of use" as a variable may appear in a software evaluation instrument, but as an isolated item, it may not provide valuable information. The audience for the software must be considered, the classroom environment in which it is to be used (will a teacher be present and supervising?) and a myriad of related factors. An understanding of the relative weakness of the item illuminates for the evaluator new considerations when making decisions.

Dudley-Marling and Owston (1987) found the reliability of some evaluation tools limited, as the evaluation instruments were not sensitive to subjectivity in respondents, requiring individuals to rate the strength of their agreement with statements such as "the software is easy to use." In other words, no specific criterion had been established for what makes software "easy to use," for example, presence or absence of help functions, intuitive icons, etc.

In addition, Dudley-Marling and Owston caution that findings from this type of assessment may be misleading; while few titles were rated as exemplary, instruments of this type draw responses that tend to follow a normal curve, making it difficult to draw inferences about the actual quality of the software.

Owston developed a criterion based scale, the York Educational Software Evaluation Scale (YESES) designed to elicit responses based on greater objectivity. Four dimensions of the software are examined: pedagogical content, the knowledge and skills the software purports teach; instructional presentation, the degree to which the software utilizes the capabilities of the computer in presenting the content; documentation, accompanying materials providing information on how to use the software from a technical and pedagogical perspective; and technical accuracy, essentially, how well the software runs.

YESES evaluates each dimension on a four-point, criterion based scale, using exemplary, desirable, minimally acceptable, and deficient as points on the scale. YESES is used by the York University Faculty of Education to conduct evaluations of educational software. Two or three teachers, trained in the use of YESES, evaluate software. Their evaluations are entered into a database.

The results of 105 YESES evaluations indicate that teachers judged only five percent of the titles to be exemplary in terms of pedagogical content and instructional presentation. Eleven percent of the packages were judged to have exemplary documentation, and nine percent were judged to be exemplary in terms of technical adequacy.

Nearly one-third of the programs were judged to be deficient on pedagogical content and instructional presentation. Thirty-four percent were judged to be minimally acceptable in terms of pedagogical content and 36 percent were rated minimally acceptable on instructional presentation. Forty-seven percent of the packages were judged to have minimally acceptable documentation and 38 percent were rated as having minimally acceptable technical adequacy. It must be remembered that these findings were arrived at using a sample of programs available in the mid 1980's. In addition, outcome assessments for student learning gains are not considered, so the actual effectiveness of the software is only inferred based on teachers' judgments on these four dimensions. However, the findings are useful to examine from a historical perspective, and because the criteria on which they are based remains applicable to current software packages, serving as a possible heuristic for software evaluation.

Further identification of the inadequacies of software evaluation instruments was presented by Gonce-Winder and Walbesser (1987). Using four software evaluation forms developed in the early 1980s as exemplars, Gonce-Winder and Walbesser examine the variance that exists among different instruments, and identify variables that pose threats to reliability and validity.

MicroSIFT (Microcomputer Software and Information for Teachers) developed two forms, one for description of software, and another for evaluation. The description form required information about subject, grade, hardware necessarily to run the program, and program operating instructions. The courseware evaluation form contained detailed questions about the content, instructional characteristics, and technical attributes. Evaluators rated software on scales of one to five and selected from among four qualifying statements for final recommendation.

The Educational Products Information Exchange Institute adapted their instructional materials analysis instrument to include software. The following variables were examined: rationale of the software; learner objectives; content; methods and approach; and means of assessment.

Courseware Report Card uses the criteria of performance - how well the program runs, ease of use, error handling, documentation, and educational value in making evaluations. A paragraph discussion accompanies each category and a final assessment in made in terms of a grade between A and E.

The National Council of Teachers of English

Committee on Instructional Technology developed a seven-page instrument which assesses presentation of content; instructional strategy for drill and practice, tutorial, simulation and/or problem solving, word processing/text editing; appropriateness to the teaching of language arts; ease of operation; appropriate documentation and support materials and program management and record keeping. Results are summarized in an overall rating sheet which also includes observations of student reactions to the software.

The weaknesses identified in the forms arise, in part, because of the nature of the language used in the forms. That is, names for variables being assessed may have different meanings for different respondents.

As stated in Dudley-Marling and Owston (1985), the likert-type scales used tend to allow for personal biases to make significant contributions to the evaluation. It could be argued, however, that even binary-structured responses, for example, "does the software contain a means of assessment?" may be influenced by subjectivity in software packages where the user must makes inferences about the existence of an attribute. For example, the means of assessment may not be a traditional "test," but rather an innovative method not recognized by the evaluator.

Other characteristics that diminish reliability include open-ended items, or an overabundance of items. Free responses must be interpreted, and may not always be interpreted accurately, and long forms are not always given the conscientious attention that shorter forms receive.

Gonce-Winder and Walbesser reiterate suggestions made by researchers with respect to characteristics of effective software.

> "(1) Capitalizing on the interactive capabilities of the computer, (2) using the individualizing capabilities of the computer, (3) using presentations that are only slightly text dependent, (4) using pictures to accompany instructions that are relevant to the substance and text of the instruction, (5) using material that has discernable educational objectives with whatever entertaining features it may possess, (6) presenting instructions for how to use the program that appear in short, comprehensive stages, (7) little dependence on auxiliary print material, (8) presenting material in a coherent instructional context, and (9) presenting material that is motivating to the student's continued attention" (p. 264).

"[F]reedom from technical errors; reflection of current and valid curricula; capitalization on the unique capabilities of the computer; use of information data bases within the program; immediate, positive, and instructional feedback; inclusion of diagnostic prescriptive and branching features in the program; flexibility for teacher modification of programs; inclusion of supplemental instructional materials" (p. 264) are suggestions credited to L. S. Golub also posed by Gonce-Winder and Walbesser.

In addition, the actual layout of the software, with respect to screen design, the nature of the feedback present in the software package, that is, feedback schedules, feedback types and function, and sensitivity to possible sexist or racist features in the package are issues for consideration.

Gonce-Winder and Walbesser (1987) propose a software evaluation form comprised of an off-line and on-line section. The off-line analysis contains items pertaining to the teacher's instructional needs and the on-line analysis contains items pertaining to the student's needs. (It can be argued that this distinction is somewhat specious; why would the teacher's needs necessarily differ from the student's needs? Presumably, they share the need for learning. In addition, why would the meeting of the teacher's needs be assessed off-line and the student's needs on-line? This is not made clear.)

The form requires the respondent to indicate presence or absence of a collection of attributes. Part I of the form, the teacher's needs section, contains the items assessing the following: software needs, cost, copyright, execution time, intended audience (by grade, age and ability level,) grouping (individual, pairs, small group,) piloting information, objective, stated purpose of the lesson, and accompanying instructional materials. The bottom of the form contains an instruction line informing the evaluator that all asterisked items need to be found present in order to continue with the evaluation. If all such items are not present, the decision not to purchase the software should be made.

The asterisked items included hardware needs, cost, software needs, piloting information, behavior objective, and stated purpose of the lesson. Presumably, the teacher reads the documentation and determines whether these criterion have been met, as this is the off-line examination. It could thus be argued that the documentation will "make or break" the software package.

If the asterisked items are present, the next step is for the evaluator to confirm the presence of the following via an on-line examination: title page, prerequisites, pretest, objective, instruction, post test, and record keeping. The instruction item is broken down into subcategories in which the evaluator responds to items such as "appropriate use of color." It could be argued that the word "appropriate" is inherently subjective; to use it in the context of a presence/absence condition is to defeat the purpose of the binary response. What is "appropriate" for one evaluator may not be for another. Again, as in part one, some items are asterisked, and in order for the software to be recommended for purchasing, all asterisked items must be evaluated as present.

It could be argued that the binary "present/absent" scale masks important data. Only a criterion which can be evaluated in terms of a yes/no answer can be evaluated, or else the criterion which really does not lend itself to that format may be distorted to fit such a framework. A package that meets instructional goals "somewhat" is either accepted unconditionally or rejected. While objectivity may be the goal, the very nature of evaluation is subjective; it is an evaluator's opinion, his or her interpretation of what he or she is examining. Perhaps the goal of software evaluation should not be pure objectivity, but greater awareness of variables that may prove important and how they function in the software. Absence or presence of color, for example, may be detrimental to the effectiveness of software in one context, but not in another, much the same way that a black and white movie can be more effective than one in color.

EPIE (Educational Products Information Exchange) Institute publishes an evaluation form along with a guide for analysts developed to be used as a guide for software selection. Responses to the multi-item questionnaire are open-ended and yes/no responses with room for explanation. The form is conceived with the intent that it be used not just for one evaluation, but that the data on the form be made available to other instructors contemplating either purchase or use of a given software package.

Sixteen pages in length, the items begin by assessing intended users and curriculum role, and move to an evaluation of the software's content, instructional intent, and learner objectives. Evaluators must cite examples from the program to substantiate their comments. The evaluation form allows for the consideration of content, methods and approach, and assessment. The form also requires evaluators to provide detailed descriptions of the general program structure (instructional approach, i.e., drill and practice, number of lessons in the package, number of items in the lesson, and approximate duration of lesson), content description and evaluation, teacher and student use description and evaluation, program management description, management system evaluation, and recommendations to the producer. Finally a 10-point scale is used for an overall evaluation.

While instruments have been criticized for their lack of control over evaluator subjectivity, the EPIE evaluation form to some extent controls for this subjectivity by requiring respondents to quote, where possible, or infer, where quoting is not possible, from the software itself when responding to many of the items. If a quotation cannot be made, it can be assumed that the respondent has inferred the particular information contained in the response. Of course, simply quoting from the documentation or software itself does not guarantee that the respondent is correct in his or her assumptions; however, it does allow for greater accountability in responses.

The software evaluation form developed by Shuell & Schueckler (1989) is less comprehensive than the open-ended instrument developed by EPIE. One strength of the EPIE form is the requirement that evaluators actually cite from the software to substantiate their opinions. Its downside is the amount of time and effort required to complete the form. In contrast to the EPIE form, Shuell's and Schueckler's instrument contains only 19 items, all of which require respondents to simply circle the appropriate point on a six point scale, with the focus on the degree to which evaluators believe that software is based on instructional principles.

It has been argued that any checklist, however, may not be adequate to convey the actual instructional power of a package (McDougal & Squires, 1995). While checklists may be useful for determining presence or absence of given features, they are unable to assess the power of the interaction of the software package and the teacher. That is, teachers may present creative approaches to the use of a software package that might have been inherent in the package, but that needed the right teacher to make use of them.

It could be argued that following this logic might allow for any instructional tool to be evaluated positively, for a talented and creative teacher can find something innovative and instructionally useful in any package, whereas a poor teacher may diminish the effectiveness of the most outstanding package. Perhaps it would be more useful to postulate that checklist evaluation forms should be considered from the perspective of reports of features, but that teachers should be aware that absence or presence of given features in isolation need not be associated with a positive or negative rating or use of the software.

Validity Issues In Software Evaluation

Useful evaluation research is sensitive to validity issues; inherent in the software evaluation process are threats to both internal and external validity. Learning is a function of multiple variables that, in a classroom, cannot be separated out. Consider, for example, a software evaluation conducted in a classroom during which students use the software package for a given amount of time and are then tested on a learning outcome. Any number of factors, some controllable by teachers, some controllable by students, could be responsible for student growth or the lack of it. Unless the number of subjects (in this case, classrooms) is fairly large, this threat to internal validity remains. In addition, small samples result in limited generalizability, as do laboratory conditions, which negatively impacts external validity.

Between group designs introduce the possibility of selection bias (Duncan, 1993) due to the nonrandom assignment of teachers to students. Characteristics of a given class may greatly impact the results of such studies, characteristics such as class performance and class interest in a subject. Even if the same instructor teaches different sections of a course, he or she may not perform identically in the given sections; interactions between instructor and students, or instructor and experimental condition may impact the results.

When matching on student characteristics is attempted, the characteristics selected are typically determined by choice of outcome measures (Duncan, 1993). For example, for evaluation of the effectiveness of a software package that is developed to increase reading achievement, subjects might be matched on reading ability and other related factors. However, this control does not contain validity threats from other sources.

Within group designs also pose threats to validity. These designs typically introduce software at different points in the course and compare the presence and absence of software conditions. Unfortunately, different topics may require different cognitive strategies, thus confounding the results of such studies.

Ransdell (1993) presents a software evaluation project that exemplifies validity threats. An introductory college psychology course was conducted in the traditional lecture format, or supplemented with computer-based course-related activities projected on a screen and developed by their professor. An honors class also used the software, but in a weekly lab, instead of as a supplement to class lectures. The first day of class students were given a short questionnaire that asked general demographic information and level of computer experience. A mid-term questionnaire on which the students evaluated the software through open-ended questions about what they liked or did not like was administered. Students did not know that the professor was the author of the software.

Responses to the evaluation questionnaire suggest that instructional method, type of student, and ways in which the software is implemented impact the perceived effectiveness of the software. (The actual effectiveness is not discernable from this study as no learning outcome assessments were performed.)

The interaction between the software and the other elements of the classroom is reflected in comments made by students to the effect that the teacher's performance was impacted by the use of the software; students felt the instructor's pace was somewhat slower. Community college students typically favored this, while honors students were somewhat disappointed with the slower pace, the simplistic nature of the program, and what they felt was a repetitive aspect of the lesson. However, honors students felt that the computer allowed the instructor to cover material in greater detail, a factor which pleased them. Students reported a greater level of class participation, and rated the computer condition of the class as being more interesting and fun.

However, a study such as this is replete with threats to internal validity and external and ecological validity. It is not possible to tease out the impact that the particular instructor has on how students perceive the software, just as it is not possible to isolate the variables within a class of students that will impact how they interact with software. For example, students using the computer alone will have different perceptions of their experience from students using it with a partner, and of course, students' perception of collaborative learning will impact on how they view the experience.

In addition, novelty effects have been shown to have a strong impact on how a given condition is perceived. Students exposed to a software package for the first time are likely to react differently from those who are very familiar with software. Ransdell (1993, p. 230) provides a list of threat to internal and external validity shown in Table 3.

Table 3

Factors That Threaten Validity

Internal Validity Factors
uneven rates of student attrition over time
improvement on exams with practice alone
student-selected class attendance, meeting times
unavoidable interactions (i.e. attendance * performance)
variable interest in topics
variable difficulty of topics
novelty, order, recency effects
necessary adaptations over time (i.e., extra credit, reviews)
definitions of independent variable (i.e. "media")
External and Ecological Validity Factors
isolating method from general teaching style
evaluation while evaluating knowledge
studies of short duration
studies conducted outside the classroom
homogeneous samples of college student ability
limited computer resources
bias of publishing "positive" results
general instructor as observer biases
general student as respondent biases

Threats to internal validity diminish the degree to which the effectiveness of a software package can be judged apart from other intervening variables. Student attrition, attendance, and participation can have a strong impact on software effectiveness; if the brightest students in a class are repeatedly absent, hypothetically, the means for learning outcomes may be lower, irrespective of the introduction of a new software package. Classroom practices, such as reviews, homework, and practice impact on student achievement, and it is often very difficult to determine the extent of their effect as compared to the use of the computer. The topic for instruction itself, that is, how difficult, or how interesting the subject is, may prove a strong determinant of student achievement. Finally, the complexity of educational software presents difficulties for researchers trying to identify the essential characteristics of the educational activity being offered; that is, it is difficult to isolate the specific factors in a software package that contribute to learning (Duncan, 1993; McDougal & Squires, 1995). Therefore, identification of the independent variable becomes a more complex, and quite challenging task. Without knowing what components are associated with successful software, it becomes very difficult to draw conclusions that can be useful in future software selection.

Likewise, threats to external validity, such as variability in teacher style, composition of student body, and duration of study impact the degree to which a software package can be credited for learning outcomes. Certainly, computer resources is a significant issue. Assessment of the effectiveness of a software package through comparing the achievement measures of a group who used it against a group who did not, typically will not allow for sufficient control of confounding variables.

Given the number of variables that interact with use of software on student achievement, it may be useful to consider using multi-variate designs in evaluating software (Duncan, 1993). These designs allow researchers to assess the amount of achievement associated with selected predictor variables, and allow for assessment of a collection of variables, such as student characteristics, features of the software, and features of the instructional environment (Duncan, 1993). An alternative strategy might be to develop an approach to software selection that considers the interactions of teacher and student and software (McDougal and Squires, 1995). That is, the approach to software selection would be one which considers the use of the software apart from its attributes. Such an approach factors in the probability that different teachers and different students will use the software differently; it may therefore be misleading to select software purely on the basis of a set of responses to questions about its features. The interaction between the presence or absence of a particular feature in the software and the use of that software in the classroom will differ from classroom to classroom. In addition, since teachers can expand upon the software, the absence of a particular feature need not mean that the software should be excluded from selection; the package may contain other excellent features and the lack of the feature in question may be made up for by the teacher (McDougal & Squires, 1995). For example, a software package that does not explicitly state its learning objective for the student may nonetheless contain other valuable attributes, and the absence of an explicit objective may prove an asset in increasing the flexibility of the software in terms of how it can be used. While the absence of this feature might prove a negative without teacher input, in the presence of teacher input, the negative may become a neutral or even positive attribute.

The weights that teachers assign peer recommendations for software become significant in view of these considerations. That is, a software package that "works" for teacher A may not be as effective for teacher B. Teacher A may have a computer for every student, while teacher B may not. Teacher A may allow students less time to interact with the software than teacher B. In short, the generalizability of results should be viewed with attention to validity threats.

Summary

Educationally sound software appears to share many of the characteristics that educationally sound, traditionally delivered instruction is associated with. Instructionally effective software conveys the learning objectives to the students, contains clear presentations appropriate for the grade and ability level of the students, uses appropriate examples, has feedback, reinforcement, provision for guided and independent practice, maintains student interest, and a provides a means of assessment students' success in meeting the intended learning objectives (Castellan, 1993; Duncan, 1993).

The relationship between educational software and traditional classroom instruction suggests that expert teachers are likely to be cognizant of those features associated with sound instruction in software packages. Several questions are suggested by this assertion: 1) Do the schemas that teachers have for evaluating the effectiveness of instruction transfer to an electronic environment? 2) Are the schemas that experts use to evaluate software different from the schemas that novices use when given the same task?

The answer to the first question might be evidenced by a commonality of responses generated by teachers on a software evaluation task. If teachers are noting the presence or absence of similar variables then it can be inferred that they are operating on a common schema. (This is not to say that this teaching schema is sufficient to judge software; it may not be. A truer test of that might be to let teachers evaluate software and then test their decisions against the achievement of actual students.) The teachers presumably are identifying the problem and employing similar approaches, probably due to similar representations of the problem. The schemas they are using can be inferred based on their responses to the software evaluation question. Likewise, the schemas that the novices possess for approaching the problem can also be inferred from their responses.

CHAPTER IV

Hypotheses

Research suggests that experts and novices have qualitatively different approaches to solving problems. These differences are, in part, associated with the individual's knowledge base; experts have larger, well organized knowledge bases and a deeper understanding of their domains than novices (Chi, Feltovich, & Glaser, 1981; Glaser, 1984). They are able to appropriately apply general laws or rules to problem situations. Novices, with a more superficial understanding of the domain, tend to focus on cursory details, often aware of abstract rules but unable to apply them. Experts are able to rely on superior pattern recognition competencies, and are able to automate many of the routine tasks novices expend processing capacity on (Larkin, et al, 1980). Their highly developed metacognitive skills allow them to monitor their progress more effectively than novices.

Mirroring the general characteristics of expertise in other domains, both well- and ill-structured, expert teachers have a larger, better organized knowledge base than novices. When performing their tasks, or, approaching a teaching problem, expert teachers are more easily able to discern the relevant information from the superficial information, and can readily identify patterns in the classroom context that are useful in guiding their course of action when approaching tasks or problems.

Software selection, while a relatively new problem in the teaching domain, requires the identification of an instructionally sound lesson or activity, a fairly common place teaching problem. The goal or solution requires recognition of the attributes of the software most closely associated with achievement of the desired learning goal.

While both teachers and novices may be able to identify the objective an educational software package should accomplish, it could be argued that the expert teacher should be better able to focus on those attributes of the software that are associated with sound pedagogical instruction than the novice. The teacher, with a deeper understanding of the problem, should be able to look beyond the superficial features of the software to the attributes most pertinent to sound instruction, and to apply the theories of what constitutes sound instruction in a way that the novice cannot. Novices would most probably not have the same schemas for effective instruction that experts have, and not attend to the same attributes in the software.

Tables 4-6 list the variables that have been identified by researchers as being significant in software evaluation. Many of these variables lie on a continuum, rather than exist in binary form. That is, there are gradations of the variable present in the software-i.e., there may be feedback, but it may not be effective; or the colors may be washed out or too vibrant and distracting.

Variables categorized as technical refer to a mechanical, audio, visual, or programming characteristic. They are considerations that the programmer would have made when developing the software, and might also be of consideration in a software package that was not intended to be educational. Table 4 provides a listing of researchers who identify given technical variables as being significant in software selection and a brief explanation of these variables.

Variables that refer to content oriented issues focus on the subject matter of the software, the specific information being presented apart from the quality of the presentation. Issues such as accuracy of the information being presented, or importance of the information being presented are contented oriented. Table 5 provides a listing of researchers who identify given content variables as being significant in software selection and describes these variables.

Variables categorized as pedagogical refer to issues in learning theory, classroom practice, or related instructional issues. Table 6 provides a listing of researchers who identify given pedagogical variables as being significant in software selection and provides brief explanations of pedagogically related variables.

Table 4

Technical Variables Identified by Researchers as Significant in Instructional Software Effectiveness

Variable	Description	Researcher
Bug Free	Absence of programming errors that would cause the software to run poorly.	EPIE (1983); Gonce-Winder & Walbesser (1987)
Consistency in Interface	Consistent use of keystrokes and icons throughout the software. For example, if the F10 key is used to clear the screen in one situation, it should be used to clear the screen throughout the software package.	EPIE (1983)
Directions On Line	Presence of instructions that users can access from within the program.	Shuell & Schueckler (1989)
Computer Capabilities	Use of the special features that are particular to computers, such as	Bitter (1993)

Used	randomization, animation, etc.	
User Controlled Navigation	User control over where in the software package to explore; there is no fixed order of events that must be followed.	EPIE (1983)
Place holders	Provision of place holders that allow users to resume their positions in the software package after quitting.	Gonce-Winder & Walbesser (1987)
Software Modification Possible	Software can be altered by teachers to accommodate their needs	Castellan (1993)
Ease of Use	Typing errors are easily erased, selections made easily.	EPIE (1983)
Color	Presence of color.	EPIE, (1983); Gonce-Winder & Walbesser (1987)
Audio	Presence of audio.	EPIE 1983

Graphics	Presence of graphics.	EPIE (1983); Schuell & Schueckler (1989)
Animation	Presence of animation.	Bitter (1993)
Cost	The price is not prohibitively expensive.	EPIE (1983); Schuell & Schueckler (1989)
Execution Time	The execution time is listed in the software.	Shuell & Schueckler (1989)
Hardware Requirements	The software can be run on most computers.	EPIE (1983); Schuell & Schueckler (1989)

Table 5

Content Variables Identified by Researchers as Significant in Instructional Software Effectiveness

Variable Name	Description	Researcher
Accuracy of Information	The degree to which the information presented in the software is accurate.	Castellan (1993)
Importance of Information	The degree to which the information presented in the software is important. The lesson should have some value, and not exist for the purpose of creating exciting animations, etc.	Castellan (1993); EPIE (1983); Gonce-Winder & Walbesser (1987)
Stereotype/Bias Free	Absence of stereotypes or biases.	EPIE (1983)
Depth of Information	Comprehensive in scope.	EPIE (1983)

Table 6

Pedagogical Variables Identified by Researchers as Significant in Instructional Software Effectiveness

Variable Name	Description	Researcher
States Learning Objectives	The learning objectives are stated in the software.	Castellan (1993); EPIE (1983)
Assesses Prerequisite Knowledge and Skills	Knowledge of information necessary to benefit from using the software is assessed. For example, a lesson on long division might assess a user's ability to do short division.	Schuell & Schueckler (1989)
Reviews Prerequisite Knowledge and Skills	Knowledge necessary to benefit from using the software is reviewed.	Schuell & Schueckler (1989)
Teaches Prerequisite Knowledge and Skills	Knowledge necessary to benefit from using the software is	Schuell & Schueckler (1989)

	taught. For example, difficult vocabulary words that appear in a passage of text are presented before users encounter the passage.	
Prescriptive	The software assesses users' weaknesses and prescribes activities to remediate the weaknesses.	Schuell & Schueckler (1989)
Grade Appropriate	The software is appropriate for the grade level of its intended users.	EPIE (1983); Gonce-Winder & Walbesser (1987)
Varying Levels of Difficulty	The activities in the software package vary in levels of difficulty. For example, a drill package in spelling might have	Gonce-Winder & Walbesser (1987); McDougal & Squires 1995)

	three levels of difficulty: a basic, intermediate, and advanced level.	
Presents Information in Appropriate Blocks	The information is presented in appropriate units, e.g., a lesson on the Civil War might have separate units for the many issues that were involved.	Schuell & Schueckler (1989)
Appropriateness of Presentation	The tone of the presentation is appropriate for the content and users, e.g., an MTV format would not be appropriate for a lesson on the Holocaust.	Castellan (1993); EPIE (1983)
Appropriate Examples Used	The examples presented are reasonable with respect to content and the users' grade and ability level, e.g., some	EPIE (1983); Gonce-Winder & Walbesser (1987)

	analogies might be lost on very young users.	
Appropriate Pacing	The pacing of the lesson is appropriate for the grade and ability level of the users.	EPIE (1983); Gonce-Winder & Walbesser (1987)
Appropriate Feedback	The feedback is pedagogically sound, e.g., there is feedback on responses, reinforcement for correct answers and for perseverance.	EPIE (1983); Gonce-Winder & Walbesser (1987)
Motivating	The software holds the attention of the user.	Schuell & Schueckler (1989)
Learner Directed	The user controls which areas of the software to go to select level of difficulty.	EPIE (1983); Gonce-Winder & Walbesser 1987); McDougal & Squires (1995)

Guided Practice	The software provides the opportunity for users to practice skills with the guidance of the program.	Schuell & Schueckler (1989)
Independent Practice	The software provides the opportunity for users to practice skills alone.	Schuell & Schueckler (1989)
Closure of Lesson	The lesson has closure, i,e., there is a summary of information presented, perhaps suggestions for further reading, etc.	Schuell & Schueckler (1989)
Appropriate Structure of Software	The structure of the software (drill and practice, tutorial, problem solving, simulation, etc.) is appropriate for the subject matter and the general	Roblyer, Edwards, & Havriluk (1997)

	presentation. For example, a drill and practice program might be appropriate for the instructional goal of learning the multiplication tables, but less appropriate for a lesson on photosynthesis.	
Means of Assessment of Learning Outcomes	There is some form of assessment built into the software, such as a quiz.	Castellan (1993); EPIE (1983); Gonce-Winder & Walbesser (1987)
Teacher Record keeping	There is a way for the teacher to keep records of students' progress, of the time they spend using the software, of modules in the software they have explored.	EPIE (1983); Gonce-Winder & Walbesser (1987)
May Be Altered by Teacher	The program can be altered by the	EPIE (1983)

	teacher to accommodate his or her learning objectives for the students. For example, the names of the characters in word problems could be changed to reflect the names of students in the class.	
May be Used Cooperatively or Alone	The program is flexible enough to be effectively used in groups or alone.	Schuell & Schueckler (1989)

Tables 4-6 provide a guide to variables identified by researchers as significant for effective software evaluation. In sum, the Technical variables help describe those issues relevant to the running of the software. The Content variables refer to those issues pertinent to the information presented in the software. The Pedagogical variables describe those issues that pertain to instruction, apart from content. That is, they describe those issues of interest to teachers and learning specialists, issues that are not necessarily associated with content.

It is of importance to remember that the categories often overlap; that is, faulty programming, or an interface that is difficult to use, may have an impact on the user's ability to use the software, and thus, impact the pedagogical variables. If the software is difficult to use, the user may not be able to take advantage of all that it might offer. That is, the user might never be exposed to the valuable skills reinforcement if the software is too difficult to use. The importance of the interaction between the categories of Technical, Content, and Pedagogical cannot be underestimated.

The expert-novice paradigm provides a framework through which researchers can examine the software evaluation strategies of experts and novices. By examining the variables identified during evaluation of a software package by experts and novice teachers, researchers are providing a window into the strategies engaged in by teachers and novices, as well as their representations of the problem of evaluation.

Software evaluation is a form of decision making, not unlike selecting candidates for admission to medical school. Those seeking to make a decision must arrive at a set of criteria against which they will match the contenders. How they select and weight the criteria may be a reflection of how they represent the problem.

This study examines how teachers who have experience in teaching and novices who have no teaching background at all differ in the way they approach the software evaluation task. However, the researcher also wishes to examine the role that experience with educational technology may have on teachers' ability to evaluate software effectively. For this purpose, three groups are used in the study: a group of highly competent teachers who have no formal educational technology background; a group of college students who have no experience in teaching; and two teachers who are highly competent in and have formal education in using technology in the classroom.

The small group of technology-experienced teachers is not included in the quantitative analysis, but rather, serves as a baseline, or model, of how teachers with formal technology training approach the software evaluation task. For the purpose of clarity, the groups are labeled, Novice, Teacher, and Educational Technologist. The Educational Technologist group is comprised of the two teachers who possess technology expertise. All teachers included in this study are believed by their supervisors and peers to be expert teachers. Thus, the study is a contrast of teacher, or experienced, and novice or inexperienced, approaches to a problem, the teachers being highly competent educators and the novices, individuals who have no teaching background or experience.

Hypotheses

The purpose of this investigation is to assess the differences, if any, in features that the technologists, teachers, and novices attend to when given the problem of evaluating software for instructional effectiveness. Although on a superficial level, the problem may be novel to teachers who have not had much experience with technology, it can be argued that on a deeper level, the problem is asking solvers to make a determination about the instructional effectiveness of a lesson, a task teachers engage in on a regular basis.

The following questions were instrumental in guiding the analysis.

1. Is there a significant difference between the groups in the percentage of responses generated that were coded as pedagogical issues, content issues, and technical issues? Is this consistent across programs?

This question is based on the assumption that teachers will have a qualitatively different representation of the problem than novices. If teachers' representation of the problem is influenced by their knowledge bases, which would contain information based on prior teaching experience, on pedagogical knowledge, as well as information sources shared by novices it follows then, that their think aloud protocols should be reflective of the representation they have for the software selection problem, and the information they are accessing to solve it, as well as the information they are focusing on in the problem itself.

2. Is there a significant difference between the groups in the number of responses generated

that were coded as Description, Explanation with No Rationale, Explanation Based on Theory or Principle, and Explanation Based on Experience or Personal Belief?

Teachers are able to apply abstract concepts appropriately when solving problems. It follows then that teachers' think aloud protocols would be reflective of these applications; that is, their protocols should contain mention of pedagogical theory or principle. Teachers are also likely to have a greater database of prior experience, which they are able to access to assist them in a problem they recognize as sharing similar attributes. It was thus expected that teachers' protocols should contain references to prior experiences.

3. For content, technical, and pedagogical issues, is there a significant difference between the groups on the distribution of responses coded as Description, Explanation with No Rationale, Explanation Based on Theory or Principle, and Explanation Based on Experience or Personal Belief? That is, for example, are there are greater number of explanations without rationale for technical statements than for pedagogical statements between the groups?

Attention to Relevant Data

Research suggests that experts tend to focus on the relevant features of a problem where novices attend to the more salient, yet often superficial features (Chi, Feltovich, & Glaser, 1981). The relevant data in a software evaluation problem is likely to include pedagogically related attributes of the software. Less relevant, though more salient features would include graphics, animation, audio, etc.

Novices are more likely to attend to these more salient characteristics, factors which may not prove as significant for teachers in determining the instructional effectiveness of a software package. That is, novices can be expected to focus on visual or audio or game-related features of the software.

Relevant features for teachers evaluating software should be associated with pedagogical issues, such as clarity of presentation, appropriateness of examples, inclusion of feedback. Teachers experienced with technology, or "educational technologists," should be particularly skilled at identifying pedagogical features in software, as they should have schemas for using computers and educational software that allows them to concentrate on educational issues, rather than having to focus on the less educationally relevant mechanical attributes of computer use.

It was hypothesized that:

H1. a) Novices, when evaluating software, will identify a greater number of variables associated with technical attributes than pedagogical attributes.

b) Teachers will identify a greater number of variables associated with learning theory and effective teaching practices (pedagogical variables) than novices.

Ability to Apply General Rules or Theories to the Problem

Experts are able to apply abstract rules or theories to problems. Educational Technologists and Teachers faced with a software evaluation problem should be better able to apply pedagogically based theories in their evaluation than novices. It was thus hypothesized that:

H2. Teachers will substantiate their reactions to the software with a greater number of explanations based on learning theory and effective teaching practices than novices.

Role of Prior Knowledge In Teaching

The role of prior knowledge in expertise has been substantiated by researchers. Teachers are likely to draw on past experiences with a problem they regard as similar to solve a novel problem.

H3. Teachers will substantiate their reactions to the software with a greater number of explanations based on prior teaching experience than novices.

CHAPTER V

Methods

Introduction

This chapter describes the research design and methods used in this study. A description of the participants, recruitment procedures, materials used in the study, and experimental methodology is included. Additionally, an explanation of the coding system is provided, including a description of the codes employed, and the methods used in data analysis.

Design

To examine the differences in schemas and strategy use between experts and novices, Wineberg (1991) examined the responses of experts (historians) with novices (high school students) to an historical evaluation problem. Sabers, Cushing, & Berliner (1991) used a similar experimental design to examine the impact of teacher expertise on the perception and monitoring of classroom events. Their study included an expert group, comprised of experienced teachers, an "advanced beginners" group, made up of individuals in a teacher training program, and a novice group, comprised of individuals with no prior teaching experience, or background, who expressed an interest in teaching. The distinction between experts, advanced beginners, and novices allowed researchers to more closely focus on the role of actual teaching experience, apart from information about teaching that was learned in a classroom, and an interest in teaching without background or experience.

The design of the present study eliminated the advanced beginners group as it is not a purpose of the study to explore the role that teacher education has in preparing individuals for the task, but rather, to be able to help to develop a model for the schemas that prototypical teachers and novices have for software evaluation. One objective of the design is to maximize the difference in expertise between the two groups. Thus, teachers were those with extensive teaching experience and novices were those without teaching experience or teaching background. A group of two participants with expertise in teaching and technology was included to provide the researcher with a standard of expertise in both areas, that is, to provide a window into the problem solving approaches of those who possess a background in both teaching and technology.

Participants

A group of 14 teachers, 14 novices, and two teachers with expertise in educational technology participated in this study.

Novices. Novices were self-selected from college students enrolled in an introductory psychology course in a large, public urban university. Five (approximately 36 percent) of the novices were male and nine (about 64 percent) were female. Novices ranged in age from 18 to 24. Three of the novices, (approximately 21 percent) were 18, six (43 percent) were 19, two (about 14 percent) were 20, two (about 14 percent) were 21 and one (about seven percent) was 24. The mean age for the novices was 19.57 (1.6 SD).

Four of the novices described themselves as Caucasian (about 29 percent); one as African American (about seven percent); three as Asian (about 21 percent); and six as Hispanic (about 43 percent).

Three (about 21 percent) reported having a high school education; eight (about 57 percent) reported having attended some college; two (approximately 14 percent) reported having an Associate's Degree; and one novice (about 7 percent) possessed a Bachelor's Degree.

A variety of majors were represented by the novices. Majors were assigned to one of five categories, Sciences, Arts, Education, Social Services, and Undecided. The major was assigned to Science if it pertained to any of the traditional sciences, including biology, chemistry, mathematics, psychology, etc. A major was assigned to Arts if it pertained to studio art, art history, English, foreign language, history, communications, etc.

Education majors were assigned to Education, and Social Services included sociology, physical therapy, etc. Undeclared majors were categorized as Undecided.

Fifty percent of the novices had Undecided majors. One (about seven percent) reported majoring in the Science disciplines; two (about 14 percent) reported majoring in one of the Arts disciplines; one (about seven percent) was an Education major; and three reported majoring in one of the Social Services disciplines.

Six (almost 43 percent) of the 14 novices reporting owning a computer; the remaining eight (about 57 percent) did not own a computer. The mean number of hours spent per week using the computer by novices was 3.71 (3.6 SD).

Six novices (almost 43 percent) reported seeing some educational software package before. The survey requested that novices list the educational software packages they had seen and this data was collapsed to indicate whether or not they had seen educational software packages. Eight novices reported seeing some educational software.

Teachers. Thirteen of the fourteen teachers were self-selected from two major urban public school systems. One teacher was self-selected from a private school. Teachers ranged in age from 29 to 59. One teacher was 29 (comprising about seven percent of the teachers;) two were 30 (about 14 percent); one was 34; one was 36; three (about 21 percent) were 39; one was 40, one was 42; two were 55; one was 56; and one was 59. Ten out of fourteen teachers were over 35 years of age. The mean age of the teachers was 42 (SD 10.05).

Since one purpose of the research was to compare the responses of teachers with novices it was important to insure that sufficient difference existed between the groups in terms of the variable of interest, that being teaching experience and ability. The researcher sought to exemplify the problem solving strategies in the unfamiliar domain of technology of prototypical good teachers; teachers who had been recognized as experienced and competent. Previous research (Leinhardt, 1986) on differences between expert and novice teachers used experience as one criterion for excellence, noting a minimum of five years as acceptable experience. Using that metric, all teachers needed to have a minimum of five years of teaching experience to qualify as having sufficient experience to participate. Additionally, it was required that they have a graduate degree. It follows then, that no teacher would be under 27 years of age.

Only two (about 14 percent) of the teachers were male. This is not surprising as in the elementary school male teachers are typically a very small minority.

Twelve (almost 86 percent) of the teachers were

Caucasian; one (about 7 percent) reported being Hispanic and the remaining teacher checked "Other" to describe ethnicity. This teacher was Jamaican American.

All fourteen teachers reported having a graduate degree. Fifty percent of the teachers taught in grades four through six; the remaining teachers taught in grades K through three. About fifty-seven percent (8 teachers) reported their specialization to be language arts; three teachers (about 21 percent) reported their specialization to be math; the remaining teachers' specializations were grouped as "other."

Three teachers (about 21 percent) reported not having a computer in their class; three reported having one machine; and seven (fifty percent) reported having two machines in their classes. One teacher is covering the computer lab this year and reported having 27 machines in the class. She had no formal educational technology training but explained that the job was offered to her and she accepted.

Ten teachers (about 71 percent) reported seeing educational software before. Twelve out of fourteen reported owning a home computer. The mean number of hours spent per week on the computer by teachers was 15.14 (SD 7.11). Nine reported seeing educational software before.

Educational Technologists. The Educational Technologist's group contained two subjects and was conceived to function as a standard of competent performance on the task. To qualify as an Educational Technologist, subjects had to possess a strong background in both teaching and instructional technology.

The Educational Technologists were 36 and 43 years of age; one male and one female; and both Caucasian. Both had education beyond a graduate degree and significant experience and educational background in educational technology. One of the technologists taught educational technology courses at a large urban university; the other was an elementary school teacher with a strong background in educational technology. Both owned their own computer and reported spending 20 hours per week using it.

Recruitment. Novices were recruited through the psychology department at a large urban public university. The psychology department organizes opportunities for their students to be participants in studies for credit in their introductory psychology course. The requirements for participation in this study were posted and students enrolled in the introductory psychology course were invited to participate.

Teachers and technologists were recruited through contacts the researcher had within a large urban public school system. The requirements for participation were distributed and those meeting the requirements were invited to participate. Teachers and technologists were offered twenty dollars for their participation. Approximately fifty percent refused to accept payment.

Human Subject Procedures. Participants received a description of the purpose of the study and what their role as a subject would entail (see Appendix B, p. 276). Participants signed a release form giving their consent to participate in the experiment(see Appendix C, p. 278). Teachers were asked to verify that they had at least five years teaching experience and excellent evaluations from their principal (See Appendix C, p. 278. Leinhardt (1986) used five years of experience and principals' evaluations as indices of expertise. Novices were asked to verify that they did not have teaching experience and did not have a teaching background (see Appendix C, p. 278).

Participants were asked to fill out a brief survey requesting biographical information (see Appendix E, p. 282) and confirming that they had the requisite experience to use a computer, and that they had not taken a formal educational technology course.

To qualify as having requisite computer experience, participants had to report that they had used a computer before and were comfortable using computers. Subjects who have never used a computer or reported strong feelings of computer anxiety were rejected. Participants who had taken a course in educational technology were rejected, as they were likely to have been trained in software selection strategies.

Materials: Software Selection

An IBM PC compatible notebook computer with multimedia capabilities was used. This machine had a Pentium with 16 megabytes of memory. The monitor displayed 256 colors. These specifications were recommended by most recently released software packages to effectively run the software. Only one computer was used to control for the effect of different hardware selection by participants.

Two mathematics software packages were selected as the stimulus, as a) mathematics is a content area with which all elementary school teachers and most novices are familiar; b) mathematics is regarded as a well structured domain (Greeno, 1976), thus the degree of subjectivity associated with content features may be diminished; and, c) a large proportion of the educational software developed is for mathematics, thus increasing the generalizability of the findings. To select a stimulus software package, a search of the EPIE database of educational software titles was conducted; evaluations of software in technology journals from 1990 to the present were reviewed; software stores were visited; and the software in the Learning Technology Department of a large urban college was examined. The EPIE database of educational software packages provides information about subject area, type of software (drill, simulation, etc.) and grade level but does not include evaluations or recommendations.

Software packages that required participants to identify nonintuitive icons, to exhibit exceptional dexterity with the mouse, or exceptional keyboarding skills were disqualified; packages that required unusually sophisticated or primitive machines were rejected; and packages that were judged to be very popular and therefore likely to be already well known by subjects were also eliminated. Programs that were obscure or atypical of most software packages were also eliminated.

Programs were matched on intended grade level, general perception of quality based on reviews in education magazines and journals, length of time needed to browse modules, and quality of graphics and sound. Two different software companies were selected to increase generalizability.

The packages selected were the Learning Company's Supersolvers Outnumbered! and Borderbund's Math Workshop. (See Appendix A, p. 274 for an overview of the software.)

These programs were found in the pilot study to be relatively easy to use with a minimum of assistance, and limited enough in scope to allow most subjects to see most of the program in the 30 minute time frame. They were colorful, contained audio, and animation. The hardware requirements were 386Sx or higher, Windows 3.1, CD ROM, speakers or headphones, and a VGA board with 256 colors at 640 x 480. These requirements were not unusual and would be accommodated by most machines.

Procedures

All participants evaluated each software package; the order in which they explored the packages was determined by randomization. Seven of the novices and seven of the teachers were randomly selected to explore Math Workshop first and Supersolvers Outnumbered second; the remaining seven novices and seven teachers explored Supersolvers Outnumbered first and Math Workshop second.

Prior to the experimental session, subjects were asked to fill out the consent form and biographical survey (see Appendix C, p. 278, and Appendix E, p. 282). Forms were examined by the experimenter to ascertain that consent had been given and that required criteria (requisite years teaching experience, principal's evaluation, etc.) had been verified.

Participants for each group met individually with the experimenter. Before the software was loaded on the computer, the researcher explained to the participants what their role in the experiment would be and gave them the written instructions (see Appendix B p. 276).

The participants were instructed to explore any components of the software that interested them for approximately 20-30 minutes, with the goal of evaluating its instructional effectiveness. They were also instructed to think aloud throughout their session with the software, providing as much detail as they felt they could offer when commenting, including any explanation that could substantiate a particular opinion. Participants were told that the researcher might prompt them to report their thoughts with the phase "What are you thinking and why" if they neglected to verbalize them. The researcher also reminded them that if they could not operate the software they were free to ask for assistance. The researcher explained that she would demonstrate briefly how to use the software before the participants actually commenced their evaluation, demonstrate the think aloud procedure, and ask the participants to practice thinking aloud. The effectiveness of the collection of think aloud protocols has been supported by Ericsson and Simon (1993) and has been used in a variety of studies (Bean, Fulmer, Zigmond & Grumet, 1995; Greeno, 1976; Hayes-Roth & Hayes-Roth, 1979; Sabers, Cushing, & Berliner, 1981; Swanson, H. L., O'Connor, J. E., & Cooney, J. B. 1990; Weinburg, 1991).

All participants were video and audio taped. They were reminded that their actions on the screen would be video taped, and that an audio tape recorder would be recording their comments as well. The video camera was positioned to the left of the participant so as to capture the screens, and the tape recorder was placed on the computer desk next to the participant.

The researcher loaded a sample software package Solitaire and used it to demonstrate the think aloud procedure. The experimenter played the game for approximately three minutes while making comments about how she thought the game was progressing, what cards she hoped would appear, and what strategies she was considering. For example, once the game was loaded, the experimenter, depending on the configuration of the cards, would say "Hmmm...I guess I'll put this Queen of Hearts on the King of Spades before I turn anything over from the pile of cards. This way, I've freed up some of the cards underneath. I know from experience this is a good strategy."

Following this demonstration, the participant practiced the think aloud procedure while playing the game. A popular game was selected as the vehicle for the think aloud demonstration so that participants could focus on the think aloud procedure without being distracted by having to learn an unfamiliar software package. Additionally, using a game as the demonstration software package instead of using an instructional software package allowed the researcher to demonstrate the think aloud procedure without exposing the participant to the researcher's software evaluation strategies.

Once the participant believed he or she understood what was expected (following approximately two to five minutes of practice), the experimenter then exited the software and the experimental software package was loaded. When the opening screen appeared on the monitor, the audio and video taping began. The procedure for the participants in all groups was the same.

Following the participant's exploration of the software packages, the researcher asked the participant to complete a brief interview with questions designed to be probes into participants' evaluation strategies (See Appendix G p. 382) and administered a survey which included a set of items which required participants to evaluate the software on a set of criteria used in the evaluation of software and pedagogical technique (see Appendix D p. 280). The interview was established to ensure that a sufficient amount of protocol data was collected for each participant in the event that a participant did not respond well to requests to think aloud, and to allow the researcher the option of being able to differentiate between probed and spontaneous responses. The survey provided a more easily quantifiable measure of how participants evaluated the software.

Coding Categories

Participants' responses were coded according to the unit of information conveyed in the response. A unit of information contains one complete thought (Ericsson & Simon, 1993). For example, the statement "The animation is pretty...but kids will find it distracting" contains two units of information, 1) "The animation is pretty" and 2) "kids will find [the animation] distracting." The screen that the participant was looking at when the protocol was made was recorded.

A two-tier coding system was established to assist in interpretation of the data. The first tier includes the major focus categories, that is, the technical, content, and pedagogical categories. The second tier includes information about the quality of the responses made by participants, that is, the type of explanation (if any) offered by the participant to defend or explain his or her initial observation. Explanations are coded as Description, Explanation with No Rationale, Explanation based on Pedagogical Theory or Principle, Explanation Based on Personal Experience, and Request for

Help or More Information. This coding system was adapted from one used in a study by Bean, Fulmer, & Zigmond (1995). Their study examines teachers' beliefs and focus when reflecting on videotapes of their own teaching.

Technical. A statement is coded Technical denoted by a T if it refers to the running of the software, the ease with which users can navigate throughout the program, how user-friendly the software was, the ease with which users could correct errors, the presence or absence of bugs, and related aspects. For the purpose of statistical analysis, these variables are organized around four clusters; Audio/visual (TA), Software (TI), Hardware (TK) and Game (TG) features. Audio/visual includes categories referring to color, audio, screen design, graphics, animation; Software includes any reference to the interface, on-line instruction on playing of game; ease of navigation throughout game, and programming issues--hardware requirements, cost, execution time, computer capabilities used, bugs, the degree to which software modification is possible. Hardware includes any reference to the user's experience of the hardware, and difficulty or ease in using the mouse, keyboard, or other features arising from the user's interaction with the hardware. Game includes any reference to the quality of the game itself, apart from a feature of the software or hardware.

This would include any reference to the degree to which the game is fun to play, how exciting or frustrating it is, what the player is about to do in terms of playing, etc.

It could be argued that the Audio/Visual category is a subcategory of Software. However, in the interest of discerning whether membership in a particular group (novice, teacher or expert) is related to attention to salience in variables, the Audio Visual category, containing the most salient features of the software, is examined separately. It can also be argued that this feature is distinct from the software category in that it is often the province of designers and artists, not necessarily programmers, and can be evaluated apart from other features of the software. Table 4 provides clusters for technical variables

Content. A statement is coded Content if it refers to the subject matter presented in the software. Such statements are distinct from pedagogical statements in that they are limited to subject matter issues and not concerned with quality of presentation with respect to instructional effectiveness. For example, the statement "I would approach the problem differently, and look at it as an estimation problem, since the numbers are so large," refers only to the content or subject matter, in this case, a math problem. In contrast, a statement like, "Breaking down the problem into steps makes is easier for students to understand. They should have set the problem up as a series of steps," refers to presentation strategies that impact the effectiveness of the instruction, and would be coded Pedagogical.

For the purpose of analysis, Content variables are organized around the following clusters: Accuracy of Information (CA); Importance of Information (CI); Stereotype or Bias Free (CS); Depth of information (CD); Problem Solving Strategies (CP); and Objective of Task (CO).

Accuracy of Information refers to statements made concerning the accuracy of the material presented in the software with respect to subject matter. For example, the answers to the math problems should be correct. Importance of Information refers to the degree to which the imformation is valuable and worthy of concentration. Stereotype or Bias Free refers to the degree in which the software refrains from using stereotypes or relaying implicit or explicit biases. For example, "All of the characters in the math game are boys--this is sending the wrong message to girls about math," would be such a statement. Depth of Information refers to the level of difficulty of the task, or the scope of the material. "Wow. This math is extremely advanced. Why, its practically algebra!" would be an example of such a statement. Statements that refer to the participants' personal problem solving strategies are coded Problem Solving Strategies. "I think I'm gonna add first and multiply later," is such an example. A statement is coded Objective of Task if it refers to the purpose of the task. For example, "This is fractions," is such a statement. Table 5 lists Content variables.

Pedagogical. A statement is coded Pedagogical if it refers to instructional issues. These include appropriateness of presentation, pacing, motivation, feedback, and related aspects. Categories are clustered for purposes of statistical analysis. Clusters include Affective issues (PA), Assessment (PS), Flexibility (in use) (PF), Quality of Presentation or Instruction (PI). Categories are clustered as Affective if they refer to motivational issues. Categories are clustered in Assessment if they refer to an assessment of the knowledge the user might need in order for the program to be effective, or if the software contained an assessment of learning outcomes after students use it. Categories are clustered in Flexibility if they refer to options for using the software in a variety of instructional settings. Categories are clustered in Quality of Presentation or Instruction if they refer to the actual presentation or teaching component of the software, the mode of presenting, the examples used, or the activities that the users would be engaged in to learn. Table 6 lists the clusters.

Other. A response is coded Other if it cannot be placed in the aforementioned categories. This category includes three types of responses. Other Description (OD) is used to describe responses participants make regarding their own experience throughout the experiment; for example "This reminds me of when I was a kid," or "I am beginning to feel tired," or "I'm not good at math." A response is coded Other Filler (OF) if it functions as filler used by the participant presumably to be able to say something to satisfy the request to think aloud. Statements such as "Wow," or "No Way," are coded as Other Filler. A statement is coded Other Noncodable (ON) if it could not be coded. Statements for which coders could not agree upon a category were coded Other Noncodable.

Explanation Categories. Within these major focus categories responses are coded as Description, Explanation with No Rationale, Explanation Based on Pedagogical Theory or Principle, Explanation Based on Experience or Personal Belief, and Request for Help or More Information. This second tier of the coding system is adapted from one that made use of the categories Description (D), Explanation with No Rationale (EN), Explanation Based on Experience or Personal Belief (EE), Explanation Based on Theory or Principle (EP) used in a study by Bean, Fulmer, & Zigmond (1995).

Statements are coded Description if the response states facts without further explanation. For example, "the screen is very colorful" would be such a statement.

Statements are coded Explanation with No Rationale if a critical observation is made but rationale for the observation is not provided. For example, "Students would find this confusing," would be such a statement, as the participant is making an evaluative statement, but not providing substantiation for the claim.

Statements are coded Explanation Based on Pedagogical Theory or Principle if the statement includes a rationale arising from an instructional theory or principle. "The software has no feedback and students do need some sort of feedback to help them assess their progress" would be an explanation based on theory or principle.

Statements are coded Explanation Based on Experience or Personal Belief if the statement includes a rationale arising from a personal belief or experience. "I've seen students get bored when the problems are repeated over and over" is such a statement.

Statements are coded Request for Help or More Information if the participants require the experimenter to explain some aspect of the software, or need help with the operation of the computer.

Transcriptions and Coding

The tapes were transcribed by the researcher using both audio and video tapes. The researcher and a graduate student specializing in educational technology and trained in coding reviewed the transcriptions and identified the individual statements within each participant's data. A statement was defined as one complete thought or idea. The researcher and the graduate student independently coded the transcriptions for each participant.

Examples of the coding can be found in transcripts of the responses for two novices, two teachers, and both technologists in Appendix F, p. 286. The teachers and novices were selected as "typical" representatives of their groups.

To increase reliability, the researcher and the graduate student each obtained a copy of the transcriptions for all participants and independently coded the statements. If they agreed on the code for a statement the code was assigned to that statement. If there was disagreement on the coding of a statement, they tried to resolve the disagreement through discussion. If the disagreement could not be resolved, the statement was coded as Other Noncodable (ON).

Data Analysis. The frequency distributions and percents of statements were obtained for all coding categories. These distributions were obtained separately for the novice group (N=14) and the teacher group (N=14) and for the combined sample. The order in which participants were presented with the stimuli was counterbalanced; that is, approximately half of the subjects in each group evaluated Math Workshop first and the other half evaluated Outnumbered first.

The data was analyzed using SPSS for Windows version 6.1. Chi Square was the statistic used for analysis. Since frequencies and percents of responses within categories do not reveal whether just a few novices or teachers are generating the majority of responses within a cell, or the responses within a cell are distributed evenly, the distribution of responses for each participant has been included. (See Appendix H p. 394).

The results reported focus on the Teacher and Novice groups. The Educational Technologist group serves as a baseline for responses generated by individuals with expertise in both domains.

Results include analysis for two software packages presented separately, and analysis for the data generated for the two software packages combined. The results for the Major Focus Categories (Technical, Content, Pedagogical) are presented first. Results for the Variable Clusters within each of the Major Focus Categories are presented next. Results for the Explanation Categories appear following each of the Major Focus Categories and their associated Variable Clusters. Figure 4 provides a conceptual overview of the analysis.

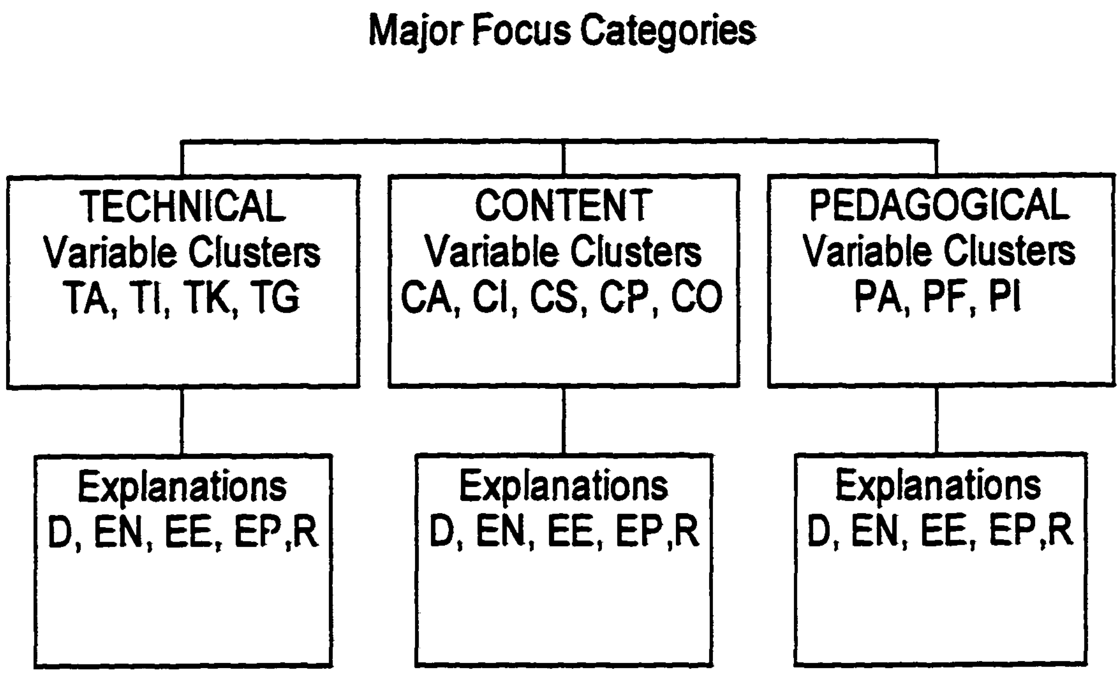

Figure 4. Conceptual Model of Data Analysis

CHAPTER VI

Results

Across the Major Focus Categories

Total Responses Generated by Teachers and Novices

Across software packages, Teachers generated more statements than Novices. When the software packages were combined Novices generated 1304 statements and Teachers generated 1519 statements. Teachers and Novices differed on the number of statements produced for each software package.

Novices produced 764 (45.7 percent) of the total number of statements for Math Workshop and 540 (46.8 percent of the total) for Outnumbered. Teachers produced 906 (54.3 percent) for Math Workshop and 613 (53.2 percent) for Outnumbered.

Teachers may have exceeded Novices in the number of statements generated because talking is a requirement of their profession, and they may have been more comfortable with talking aloud with the experimenter than the Novices.

It is not surprising that both groups generated more responses for Math Workshop than Outnumbered. Math Workshop is comprised of multiple games providing many opportunities for commentary. Additionally, Outnumbered is timed and depending on the player, could end relatively quickly compared with Math Workshop which does not end until the player terminates the game.

Novices generated a higher percentage of unrelated task statements ("Other") than Teachers. Of the statements generated by Novices, almost 33 percent for Math Workshop and about 35 percent for Outnumbered were unrelated to the task. Of the statements generated by Teachers, almost 20 percent for Math Workshop and 19 percent for Outnumbered were unrelated to the task. See Table 7 for a breakdown of the distribution of statements generated in each of the major focus categories and in the Other categories.

This finding lends support for the contention that Teachers, being expert in the domain of education evaluation, are better able to focus on the task than Novices. That is, a greater percentage of their comments are task-relevant, suggesting they are focusing on the task to a greater extent than Novices.

Summary

A preliminary examination of findings indicate that Teachers generate more statements than Novices, and that Teachers generate more task-relevant statements than

Novices. The following sections will examine results in the Major Focus Categories and Explanation categories with an emphasis on the proposed hypotheses.

Table 7

Frequencies and Percents of Statements Generated By Novices and Teachers In Major Focus Categories

	Major Focus Categories													
	T		C		P		OD		OF		ON		Totals	
Group	n	%	n	%	n	%	n	%	n	%	n	%	n	%
[a]N	282	36.9	147	19.2	85	11.1	52	6.8	139	18.2	59	7.7	764	45.7
[a]T	470	51.9	94	10.4	165	18.2	35	3.9	95	10.5	47	5.2	906	54.3
[b]N	211	39.1	93	17.2	45	8.3	47	8.7	103	19.1	41	7.6	540	46.8
[b]T	346	56.4	75	12.2	73	11.9	26	4.2	76	12.4	17	2.8	613	53.2

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. T = Technical; C = Content; P = Pedagogical; OD = Other Descriptive; OF = Other Filler; ON = Other Noncodable. Totals refers to total number of statements generated by the group.

$p > .01$.

Major Focus Categories For Each Software Package

Hypothesis 1. a) Novices, when evaluating software, will identify a greater number of variables associated with technical attributes than pedagogical attributes.

b) Teachers will identify a greater number of variables associated with learning theory and effective teaching practices (pedagogical variables) than novices.

Findings. There were significant differences between the Teachers and Novices at the .01 level on the major focus categories Technical, Content, and Pedagogical, for Math Workshop and Outnumbered. This is the case whether or not software is combined and whether or not the Other category is included in the analysis (see Table 7; Appendix H, p. 384.)

Contrary to what was expected, Novices did not generate more technical statements than Teachers for either software package. Teachers generated 470 Technical statements (51.9 percent of the total number of statements they generated) for Math Workshop. Novices generated 282 Technical statements (36.9 percent of the number of statements they generated) for Math Workshop. The pattern is repeated in Outnumbered with Teachers generating 346 statements (56.4 percent of the total number of statements they generated) and Novices generating only 211 (39.1 percent of the total number of statements they generated). See Table 7 for a distribution of the statements across the Major Focus Categories.

There is support for the assumption that Teachers would generate a greater number of Pedagogical statements than Novices. For Pedagogical statements, in Math Workshop Novices generated 85 statements (11.1 percent of the total number of statements they generated) while Teachers generated 165 statements (18.2 percent of the total number of statements they generated.) A similar pattern is seen in Outnumbered with Novices generating 45 statements (8.3 percent of the total number of statements they generated) and Teachers generating 73 statements (11.9 percent of the total number of statements they generated.)

While Teachers exceeded Novices in the percentage of statements generated for both the Technical and Pedagogical Major Focus Categories, the reverse is true for the Content category. For Content statements in Math Workshop, Novices generated 147 statements (19.2 percent of the total number of statements they generated) and Teachers generated 94 statements (10.4 percent of the total statements they generated). For Outnumbered, Novices generated 93 statements (17.2 percent of the statements they generated) and Teachers generated 75 statements (12.2 percent of the total number of statements they generated.)

Explanation Statements Across Major Focus Groups

Hypothesis 2. Teachers will substantiate their reactions to the software with a greater number of explanations based on learning theory and effective teaching practices than novices.

Hypothesis 3. Teachers will substantiate their reactions to the software with a greater number of explanations based on prior teaching experience than novices.

Findings. The difference between the Teachers and Novices is significant at the .01 level on the number of explanation statements generated for the software packages examined individually (see Table 8). This is also true when the software packages are combined.

It was hypothesized that Teachers would generate a greater percentage of Explanations Based and Experience or Personal Belief, and Explanations Based on Theory or Principle statements than Novices. These findings do support these hypotheses, although there is stronger support for teachers outperforming Novices in the Explanations that were generated based on Experience or

Personal Belief. Eleven and a half percent of the Explanation statements generated by Teachers for Math Workshop were generated in the category Explanation Based on Experience or Personal Belief. Novices generated approximately nine percent of their Explanation statements in this category. The difference between the groups is more apparent for Outnumbered. Over nine and one half percent of the Explanation statements generated by Teachers were Based On Prior Experience or Personal Belief. Novices generated 4.4 percent of Explanation statements in this category.

Neither Teachers nor Novices appear to be relying heavily on theoretical knowledge in the software evaluation task, although Teachers did exceed Novices in their generation of Explanations Based on Theory or Principle. For Math Workshop, Teachers generated .9 percent of their statements compared with Novices' .3 percent in the category of Explanations Based on Theory or Principle. For Outnumbered, Teachers' statements in the Explanations Based on Theory or Principle category comprised one percent of their Explanation statements and no statements were generated in this category by Novices.

Teachers generated a greater percentage of Explanation With No Rationale statements. Novices exceeded Teachers on the percentage of statements they generated in the Request for Help or More Information category.

The majority of Explanation statements were generated in the Description category for both groups for both software packages, with Teachers exceeding Novices in the percentage of statements generated. It can be argued that Description statements are perhaps the most easily generated, with the exception of the Other category, in that they do not require any critical assessment.

Table 8

<u>Frequencies and Percents of Explanation Statements Generated by Novices and Teachers Across Major Focus Categories</u>

	Explanation Categories													
	D		EN		EE		EP		R		Other		Totals	
Group	N	%	n	%	n	%	n	%	n	%	n	%	n	%
[a]N	249	32.6	80	10.5	70	9.2	2	.3	113	14.8	250	32.7	764	45.7
[a]T	389	42.9	115	12.7	104	11.5	8	.9	113	12.5	177	19.5	906	54.3
[b]N	196	36.3	32	5.9	24	4.4			97	18.0	191	35.4	540	46.8
[b]T	323	52.7	45	7.3	59	9.6	6	1.0	61	10.0	119	19.4	613	53.2

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. D = Description; EN = Explanation With No Rationale; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle; R = Request for Help or More Information.

$p < .01$.

<u>Within the Major Focus Groups</u>

<u>Technical</u>

Teachers generated a greater percentage of <u>Technical</u> statements than Novices. Within the <u>Technical</u> category, there was a significant difference between the groups on the variable clusters. This was true for both Software packages.

Differences between the groups on the variable clusters are consistent for both software packages, but are more apparent depending on the particular software package examined. For example, Novices generated almost twice the percentage of <u>Audio Visual</u> (TA) statements as Teachers for <u>Math Workshop</u> (11 vs. five percent) but the difference between the percentage of <u>Audio Visual</u> statements generated by the groups for Outnumbered is relatively minor by comparison (four vs. three). (See Table 9). This finding supports the hypothesis that Novices will be more attentive to the most salient features of the software evaluation problem. <u>Audio Visual</u> features of software are among the most salient.

For <u>Outnumbered</u>, Teachers generated 12 percent of their Technical statements under the <u>Software</u> (TI) category; Novices generated two percent of their <u>Technical</u> statements in this category. For <u>Math</u>

Workshop, the difference is somewhat less, with nine percent for Teachers and six percent for Novices.

Teachers generated considerably more Hardware (TK) statements than Novices. The difference between the groups is more pronounced for Math Workshop than Outnumbered.

Both groups generated the highest percentage of statements in the Game-related (TG) category, with Novices generating more Game-related statements than Teachers. This difference is minor for Math Workshop (82 for Novices vs. 81 for Teachers) but is more apparent for Outnumbered (91 for Novices vs. 81) for Teachers.

For a more detailed breakdown of response distribution, see Appendix H p. 388.

Table 9

Frequencies and Percents of Technical Statements Generated by Novices and Teachers Within Variable Clusters

Group	Technical Variable Clusters							
	TA		TI		TK		TG	
	n	%	n	%	n	%	n	%
[a]N	30	10.6	17	6.0	3	1.1	232	82.3
[a]T	25	5.3	42	8.9	20	4.3	383	81.5
[b]N	9	4.3	5	2.4	6	2.8	191	90.5
[b]T	11	3.2	40	11.6	16	4.6	279	80.6

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. TA = Audio/Visual; TI = Software Related; TK = Hardware Related; TG = Game Related.

$p > .01$.

Explanation Technical Statements

The difference between the Teachers and Novices is significant at the .01 level on the number of Explanation statements generated within the Technical category. (See Table 10). This is true when the software packages are examined separately and combined.

For both groups in both software packages, the highest percentage of Explanation statements generated under the Description (D) category, followed by the Request for Help or More Information (R) category. Teachers exceeded Novices in the percentage of Description statements generated; Novices exceeded Teachers in their production of Request for Help or More Information statements.

There is very little difference between the groups on the percentage of Explanation With No Rationale (EN) statements generated for either software package.

Teachers exceeded Novices in their production of Explanation statements Based on Experience or Personal Belief (EE) for Outnumbered (8 percent vs. .9 percent) but not for Math Workshop where 9 percent of Novices' Explanation statements and 5 percent of Teachers' statements were Based on Experience or Personal Belief.

Neither group generated any Explanation statements Based on Pedagogical Theory or Principle.

Summary. Teachers generated more Software- and Hardware-related statements than Novices; Novices exceeded Teachers on the number of Audio/Visual and Game-related statements generated. This finding is consistent with the assumption that Novices will be more attentive to salient features of the software evaluation problem than Teachers. Teachers generated a greater percentage of Description (D) statements than Novices. Novices generated a greater percentage of Request for Help or More Information statements than Teachers.

Table 10

Frequencies and Percents of Explanation Statements Generated by Novices and Teachers Across Technical Variable Clusters

	Explanation Categories									
	D		EN		EE		EP		R	
Group	n	%	n	%	n	%	n	%	N	%
[a]N	130	46.1	37	13.1	24	8.5			91	32.3
[a]T	290	61.7	60	12.8	25	5.3			95	20.2
[b]N	112	53.1	16	7.6	2	.9			81	38.4
[b]T	243	70.2	27	7.8	26	7.5			50	14.5

Notes. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. D = Description; EN = Explanation With No Rationale; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle; R = Request for Help or More Information.

$p < .01$.

Content

Differences between Teachers and Novices are significant at the .01 level for variable clusters within the Content Category for Math Workshop, but not for Outnumbered.

The largest difference between the groups occurs in Math Workshop for the Problem Solving (CP) variable cluster. Teachers generated 16 percent of their Content statements in the Problem Solving variable cluster; Novices generated 57 percent of their Content statements in this cluster. This finding supports the premise that Novices focus on the more salient issues of a problem. The Problem Solving (CP) variable cluster refers to a participant's involvement with the tasks present in the software, i.e., their statements that refer to their experiences solving the math problems in the software. These tasks are among the more salient features of the evaluation problem, and yet they do not, in and of themselves, provide information pertinent to the instructional merit of the software, which is the problem to be solved. That is, focusing on the strategies one intends to employ to solve a math problem is not the same as deciding if the math problem contributes to the educational effectiveness of the software.

A difference between the groups is also evident for both the Depth of Information (CD) and Objective of Task (CO) variable clusters. Teachers generated about twice the percentage of Depth of Information statements as Novices (26 percent vs. 12 percent) for Math Workshop, and about three percent more than Novices for Outnumbered. They also exceeded Novices in the percentage of Objective of Task (CO) statements (about 55 percent vs. 31 percent).

Neither group generated Accuracy of Information (CA) statements in Math Workshop, and no Importance of Information (CI) statements or statements dealing with Stereotypes or Biases (CS) were generated by either group for Outnumbered.

Table 11

Frequencies and Percents of Content Statements Generated by Novices and Teachers Within Variable Clusters

	Content Variable Clusters											
	CA		CI		CS		CD		CP		CO	
Group	N	%	n	%	n	%	N	%	N	%	N	%
[a]*N							18	12.2	84	57.1	45	30.6
[a]*T			2	2.1	1	1.1	24	25.5	15	16.0	52	55.3
[b]N							6	6.5	66	71.0	21	22.6
[b]T	1	1.3					6	8.0	52	69.3	16	21.3

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. CA = Accuracy of Information; CI = Importance of Information; CS = Stereotype/Bias Free; CD = Depth of Information; CP = Problem Solving Strategies; CO = Objective of Task.

*$p < .01$.

Explanation Content Statements

There was no significant difference between the Novices and Teachers on the number of Explanation statements generated for either software package whether the software was examined separately or combined and variable clusters were combined. There was a significant difference between the groups at the .01 level for Math Workshop, but not for Outnumbered when variable clusters were not combined. (See Appendix H p. 387).

Over seventy-five percent of the Explanation statements for both Teachers and Novices were Description (D) statements. Neither group generated more than six percent of their Explanations Based either on Experience or Beliefs (EE) or on Theory or Principle (EP).

Summary. The difference between groups on statements generated in Content Variable Clusters was significant for Math Workshop but not for Outnumbered. Novices exceeded Teachers in the percentage of statements concerned with the Problem Solving (CP) strategies they employed while working on the tasks in the software for Math Workshop. Teachers generated a greater percentage of Depth of Information (CD) and Objective of Task (CO) statements for Math Workshop. There was no significant difference between groups for Explanation Statements.

Table 12

Frequencies and Percents of Explanation Statements Generated by Novices and Teachers Across Content Variable Clusters

	Explanation Categories									
	D		EN		EE		EP		R	
Group	n	%	N	%	n	%	n	%	N	%
[a]N	111	75.5	21	14.3	5	3.4			10	6.8
[a]T	70	74.5	15	16.0	6	6.4	1	1.1	2	2.1
[b]N	78	83.9	3	3.2	3	3.2			9	9.7
[b]T	69	92.0	2	2.7	1	1.3			3	4.0

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. D = Description; EN = Explanation With No Rationale; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle; R = Request for Help or More Information. There were no significant differences between Teachers and Novices.

Pedagogical Statements

There were no significant differences between Teachers and Novices in the percentages of statements generated in the variable clusters for the Pedagogical category. This is true when software is combined or examined separately.

The majority of statements for Novices and Teachers were generated in the Instruction (PI) variable cluster. For Math Workshop, Novices generated 58 percent and Teachers generated 57 percent of their Pedagogical statements in this variable cluster. For Outnumbered, Novices generated 51 percent and Teachers generated 63 percent.

The Affective (PA) attribute of the software was also an area of focus for both Teachers and Novices. For Math Workshop forty-one percent of the Novices' Pedagogical statements and 38 percent of the Teachers' Pedagogical statements were generated in the Affective (PA) variable cluster. This pattern holds true for Outnumbered. Novices generated 47 percent and Teachers generated 36 percent of their Pedagogical statements in the Affective (PA) variable cluster.

Neither group concentrated heavily on Flexibility issues, with Novices' statements comprising one percent and Teachers' statements accounting for five percent of their Pedagogical statements. For Outnumbered Novices generated two percent and Teachers generated one percent of their Pedagogical statements in this variable cluster.

Table 13

Frequencies and Percents of Pedagogical Statements Generated by Novices and Teachers Within Variable Clusters

Group	Pedagogical Variable Clusters					
	PA		PF		PI	
	n	%	n	%	n	%
[a]N	35	41.2	1	1.2	49	57.6
[a]T	63	38.2	8	4.8	94	57.0
[b]N	21	46.7	1	2.2	23	51.1
[b]T	26	35.6	1	1.4	46	63.0

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. PA = Affective; PF = Flexibility; PI = Instruction.

No significant differences between groups were found on statements generated within variable clusters.

Explanation Pedagogical Statements

There was no significant difference between the Novice and Teacher groups in the number of Explanation statements generated in the Pedagogical category for either software package (see Table 14). When software is combined and variable clusters are expanded, there is significance at the .05 level. However, when software is taken individually and variable clusters are expanded, there is no significant difference between groups. (See Appendix H p. 392)

The percentage of Explanation statements based on Experience or Personal Belief (EE) exceeds Description statements for both Teachers and Novices. Explanations Based on Experience or Personal Belief in the Pedagogical category comprised 48 percent of the Novices' Explanation statements and 44 percent of the Teachers' Explanation statements for Math Workshop, and 42 percent of the Novices' and 44 percent of the Teachers' Explanation statements in Outnumbered. The distribution of Explanation statements in the Pedagogical category departs from the pattern seen in Technical and Content categories. (Explanation statements Based on Experience or Personal Belief accounted for no more than 12 percent of the Technical statements and no more than six percent of the Content statements for either software package.)

Teachers generated as much as 8 percent of their Explanation statements Based on Theory of Principle. Novices generated no statements in this category for Outnumbered, and 2 percent in Math Workshop.

Summary. There was no significant difference between the Novices and Teachers on the Explanation statements generated in the Pedagogical category. Of interest, however, is the finding that Explanation Statements Based on Experience or Personal Belief comprised as many as 48 percent of the Novices' Explanation statements and 44 percent of the Teachers' Explanation statements. This suggests that when faced with pedagogical issues, both Novices and Teachers draw upon prior experience or belief systems to understand their significance. In a broader sense, they are both activating schemas for the software evaluation problem that involve retrieving what they believe to be true about instruction, when making a judgment about effectiveness. Teachers, more than Novices, appear to be relying on theoretical principles when faced with the software evaluation task. However, the percentage of Explanations Based on Theory or Principle when compared with the bases for any of the other Explanations is relatively small, suggesting that Teachers are not employing their theoretical knowledge as much as they are other strategies.

Table 14

Frequencies and Percents of Explanation Statements Generated by Novices and Teachers Across Variable Clusters

	Explanation Categories									
	D		EN		EE		EP		R	
Group	N	%	N	%	n	%	n	%	N	%
aN	8	9.4	22	25.9	41	48.2	2	2.4	12	14.1
aT	29	17.6	40	24.2	73	44.2	7	4.2	16	9.7
bN	6	13.3	13	28.9	19	42.2			7	15.6
bT	11	15.1	16	21.9	32	43.8	6	8.2	8	11.0

Note. N = Novices; T = Teachers; a = Math Workshop; b = Outnumbered. D = Description; EN = Explanation With No Rationale; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle; R = Request for Help or More Information.
No significant differences between groups were found on statements generated within variable clusters.

Summary

For both software packages, Teachers generated a greater percentage of Technical and Pedagogical statements, and a greater percentage of Explanation Statements Based on Experience or Personal Belief and Explanation Statements Based on Theory or Principle. However, while Teachers are generating more Pedagogical statements than Novices, and more Explanations grounded in experience or theory, it is important to note the degree to which they differ from Novices. Table 15 for a summary of the percentages of statements generated by Teachers and Novices in the Major Focus Categories and for Explanations Based on Experience or Personal Belief and Explanations Based on Theory or Principle.

The following hypotheses were supported:

H1b) Teachers will identify a greater number of variables associated with learning theory and effective teaching practices (pedagogical variables) than novices.

H2. Teachers will substantiate their reactions to the software with a greater number of explanations based on learning theory and effective teaching practices than novices.

H3. Teachers will substantiate their reactions to the software with a greater number of explanations based on prior teaching experience than novices.

The hypothesis H1a) "Novices, when evaluating software, will identify a greater number of variables associated with technical attributes than pedagogical attributes" was not supported.

Table 15

Summary of Percentages of Statements Generated by Novices and Teachers In Major Focus Categories and Explanations Based on Experience or Theory

	T		C		P		EE		EP	
	MW	ON	MW	ON	MW	ON	MW	ON	MW	ON
Novices	37	39	19	17	11	8	9	4	.3	
Teachers	52	56	10	12	18	11	12	10	.9	1

Note. MW = Math Workshop; ON = Outnumbered. T = Technical; C = Content; P = Pedagogical; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle. All percentages have been rounded to the nearest tenth.

Additional Analysis

Rating of Software

All participants in the study filled out a brief questionnaire in which they were asked to evaluate specific variables using a five point likert scale. (See Appendix D p. 280 for Survey.)

Few items yielded a significant difference in the percentage of responses generated between the Novices and Teachers. Item 9, "The software provides an opportunity for students to develop and use problem solving skills," is the only item on which Teachers and Novices significantly differed for both software packages. Significant differences appear at the .05 level for Math Workshop on items 1) The information presented in the software is accurate; 5) Students would be motivated to use the software; 6) The software contains useful feedback on students' responses to questions or problems; and for Outnumbered on items 8) The software contains a means of assessment of students' progress; 10) The software is largely "drill and practice," that is, students practice specific skills, for example, their multiplication tables; and 15) The software is relatively easy to use. (See Table 16 for distribution of responses on this survey for those items which indicated significant differences between the groups.)

Table 16

Distribution of Percentage of Responses on Survey Items Indicating Significant Difference Between Teachers and Novices

	Strongly Agree		Somewhat Agree		Somewhat Disagree		Strongly Disagree		No Opinion	
Item	N	T	N	T	N	T	N	T	N	T
[a]1	100	57		36		7				
[a]5	79	29	21	57		14				
[a]6	50	14	50	50		29		7		
[b]8	43		36	43	7	50	7	7	7	
[a]9	79	21	21	79						
[b]9	64	14	29	71	7			7		7
[b]10	57	7	21	43		36	14	14	7	
[b]15	43	7	7	29	43	21	7	43		

Note. N = Novices; T = Teachers. a = Math Workshop; b = Outnumbered.

$p. < .05$.

When asked to evaluate the software on a scale of 1 to 10, 1 being most instructionally effective, Novices rated both Math Workshop and Outnumbered higher than Teachers or Educational Technologists. Educational Technologists gave both packages a much lower rating than Novices or Teachers. The mean rating on a scale of one to 10, 10 being highest, of Math Workshop was: Novices 8.8 (SD .9); Teachers 6.8, (SD 2.0); and Educational Technologists 2.5, (SD .7). For Outnumbered the mean ratings were: Novices 7.0 (SD 3.0); Teachers 5.0, (SD 2.3); and Educational Technologists 1.5 (SD .7)

Novices appear to be the least critical of the packages and Educational Technologists most critical. It could be inferred that Novices, who do not possess the schemas for features associated with educational merit are more likely to be positive in their evaluation—they do not have the tools to find fault with it and are only able to appreciate it based on its interactive or game-like qualities. Teachers, more capable than Novices in determining instructional effectiveness were able to be more critical, but it was only the Educational Technologists, with their background in both teaching and technology, who were able to examine the software and find it lacking.

Results From Educational Technologists

The function of the educational technologists group was to provide an archetype of those skilled in both teaching and technology domains. While teachers presumably possess the pedagogical knowledge needed to evaluate instruction, it is not clear whether this knowledge transfers to an electronic platform. A comparison of the schemas for teachers with and without technology training may yield some insight into this question. Thus, the technologists provide a window into the evaluation schemas and strategies of teachers who are highly competent in using educational technology.

Frequencies and percents of statements generated by Educational Technologists during the experimental task serve as a baseline for that those competent in both teaching and technology and appear in Appendix H p. 409.

For Math Workshop, the difference between the percentage of Pedagogical statements generated by Educational Technologists and Teachers is relatively small: Teachers generated 18.2 percent compared with Educational Technologists' 19 percent. However, for Outnumbered, Teachers generated 11.9 percent of their total statements in the Pedagogical category compared with 25 percent for Educational Technologists. Outnumbered is the more complicated program game-wise, with a larger number of distracters from the perspective of the problem to be solved. These distracters could be anything from animation to complicated game rules. As the examples below indicate, Educational Technologists were not as vulnerable to these distracting elements as Teachers or Novices. Examination of the protocol data generated by Novices, Teachers, and Educational Technologists lends support for the distinction between schemas held by the groups.

Attention to Pedagogical Issues

Both Educational Technologists focused entirely on the problem of software evaluation, drawing heavily on their beliefs about effective instruction. While Teachers also concentrated on some pedagogical features, the Educational Technologists' attention was grounded in being able to answer the question "Is this software instructionally effective and why?"

> "...I have a hard time with drill and practice because I think this is -- there's not a whole lot of thinking that has to go into it, there's not a whole lot of..." 30E Appendix F p. 345

"...So many of these are to use a cliche, edutainment. And the thing is, what I find very interesting there's a lot of talking and a lot of things to remember about how to run this, so it makes you think this is for older children, but the problems are on a first grade level..."

30T Appendix F p. 349

"...Now three screen in a row where the answers been in the same box. The right answer's in the same box, I could just keep clicking the same box. I wonder if that's a pattern, I wonder if they put all the right answers in the same box. That's interesting..."

29T Appendix F p. 361

"...Oh that's not really clear. Because if you're really not familiar with that function, it sounds like explanation not estimation..."

29T Appendix F p. 362

"...So now they give you a calculator

to use. I'm not sure I'm happy with that. I'm not sure I'm happy with that at all because I wonder if that really helps kids invent new ways to figure out their own solutions, I think it just does it for them….Cause I would almost, maybe if this thing gave you enough time I'd almost rather they have manipulatives there, so that they could figure it out using some blocks or whatever it is or even on their fingers if they have to, not to have some other machine do it for them. I don't think that teaches them anything…"

29T Appendix F p. 368

Qualitative Analysis: Premises Held By Novices, Teachers, and Educational Technologists

A richer understanding of the problem solving approaches taken by Teachers and Novices may be had by examining the protocol data (see Appendix F. p. 286 Some examples of participants' statements are not in Appendix F. For those quotes, page number refers to a page in the original transcription). Novices, Teachers, and Educational Technologist's problem solving approaches appear to be guided by different premises.

Beliefs About Exposure To Information and Learning

Novices harbored the belief that exposure to information was a causal agent in learning. Hence, they believed that anything leading to longer exposure was an educational asset, as the more the student would play with the game, the more significant their learning outcome would be. Thus, the affective aspect of the game was a factor they focused on (Novices generated more Audio/Visual (TA) statements than Teachers, particularly for Outnumbered, the package most complex, in terms of the playing of the game) since if the game was enjoyable to play, and students were motivated to keep playing, exposure time would increase, resulting in greater learning gains.

> "...For most kids, if they actually see math as being fun, they start to realize hey, maybe I should start playing more and they get skills in math."
>
> Participant 1N Appendix F p.322
>
> "...if they play for a longer period

of time, it would be like uhm, a conscious thing, whatever they do, it would assimilate what they've learned to what they experienced."

Participant 2N p. 13

"...because if it repeats over and over, they'll already know the answer and they'll have an idea what to do..."

Participant 13N p. 134

"...This can actually help kids cause its funny...they're playing, but they're actually learning, right?..."

Participant 12N p. 126

Neither Teachers nor Educational Technologists shared with Novices the belief that exposure to material was a causal agent in learning. Although some remarked that their students would enjoy the software, no teacher expressed the belief that prolonged exposure leads to learning gains.

Beliefs About Feedback and Learning

Teachers and Educational Technologists sought to discover how the program responded to incorrect answers. Novices, in contrast, were not as concerned with this pedagogical attribute. Novices, on average, were not troubled by instances where answers were immediately provided to students who could not correctly solve problems. Teachers and Educational Technologists viewed this unfavorably.

> "...Now I got the right answer? Does it tell you if it's the right answer? It doesn't tell you anything here. Ok if it's the right answer, it doesn't tell you. It doesn't tell you if you're correct, it doesn't tell you if you did the right thing, it doesn't tell you if you did the operation, it doesn't tell you anything. It doesn't give you information. In fact, it doesn't give you any feedback..."
>
> Participant 19T p. 197
>
> "Let's see what happens if I make a mistake."
>
> Participant 30T Appendix F p. 344

Beliefs About Prerequisite Information

Teachers also mentioned prerequisite skills that had to be in place for the games to be effective. Novices made no mention of this.

> "Can I ask you a question? If a kid can't read, how does he manage this? .... He can't play at all so this discriminates against a poor reader who might be good at math."
> 22T p. 228

Attention to Salient Features

Previous research (Greeno, 1976; Chi, Feltovich, and Glaser, 1981) indicates that when individuals encounter unfamiliar domains, they tend to be attentive to the most salient features of the problem, rather than the less obvious issues. Presumably, the degree of competence in the domain of education and in educational technology impacted participants' schemas for the task. Novices seem to focus to a greater degree than Teachers or Educational Technologists on salient features of the software evaluation problem.

> Audio/Visual (TA)
> "...The whole thing I like about these, these games is that its animated..."
> 6N Appendix F p. 338
> Problem Solving (CO)
> "...Uhm. Maybe I could put that there.

I can't find where this piece should go. Uh...Seems like everything is fitting together perfectly but then I just can't get that one piece in...leave that there for a second. Oh God!"

3N p. 33

Novices frequently referred to the lack of relevance of the tasks in the software to the math. They appeared to note only the most salient mathematical tasks as being relevant to math; they did not recognized less obvious math learning objectives. This was not true for most Teachers or the Educational Technologists.

"...This is no help for math... What kind of math is this? I have no idea..."

4N p. 40

Familiarity with Educational Software

Teachers and Educational Technologists, to a greater extent than Novices, expressed concern over software issues, specifically, their belief that the rules of the games were not intuitive. Many teachers and one of the Educational Technologists insisted on not being taught the games by the researcher, so they could better assess how intuitive they were.

Teachers generated more TI (software oriented) statements for both software packages than novices. They seemed attentive to the ease with which they could make sense of how to play, how easy it was to learn the rules, how intuitive the game was. For both teachers and Educational Technologists, a game that could not be intuitively grasped earned negative feedback.

> "...and just having to do the whole...making him walk is a distraction. ...I don't want to work that hard to get a math problem, you know."
>
> 15T p. 152
>
> "...I can't see, you know, its totally convoluted. I mean, its not clear, its not simple, and its not self explanatory. I don't like it. I mean I think not that things should always be easy, but your figuring out how to play the game is an obstacle to learning what you're supposed to learn."
>
> 29E Appendix F p.364

"...but again, I'm spending more time on this stuff, on the mechanics, than the educational value of the whole thing..."
17T p. 175

Value of the Game

Novices, to a greater extent than teachers and Educational Technologists, tended to become involved in the game itself, losing sight of the problem at hand, that being to evaluate how instructionally effective the game was.

While Teachers and Educational Technologists occasionally found themselves caught up in the games, they tended to be attentive to the fact that overly complex games were often an educational liability.

"...but that's the problem I think, there was too much going on all at once. It was less educational..."
20T p. 202

"...To me its taking too long to get to the problem, finding the problem, you know..."
15T p. 157

"...But while the game itself is too

> confusing and its too frustrating to get to the word problems to make it worth while..."
>
> Participant 25T p. 252

The Educational Technologists spent relatively little time on games, often not even bothering to complete them once they understood how they were played. They appeared to have a schema for how the games worked, and thus, less cognitive capacity had to be devoted to the process of navigating throughout the game. Educational Technologists frequently referred to other educational games, using these games and their experiences with the games as a basis for comparison. In completely familiar territory, their cognitive capacities were not expended on how to navigate throughout the software, or on the basic mathematical knowledge needed to solve some of the games.

> "...This is interesting because...you ever play the game Moncala? Moncala is a similar...its in a little tray and there are ten little pockets in each and you have you start with four little stones in each one and the idea is to clear your

side of the tray before the other
person.... "
30E Appendix p.346
"... that [Oregon Trail] didn't explain
to him what to do, and it wasn't
until Oregon Trail 2 or Oregon Trail
3 that they tried to actually address
those problems...
you figure in a class...usually they
have a schedule for which day the kid
gets the computer...so Johnny spends 45
minutes ambling through this one
program, either getting frustrated
and turning it down and going on to
something else, or not getting
frustrated and just sitting and
playing..."
29E Appendix F p.372

Familiarity with educational software as well as a solid knowledge base in education allowed them to devote themselves to the task as hand, namely, evaluating the software for instructional merit.

Interview

The brief interview was included in the study to allow the researcher to offer specific probes that might elicit responses not spontaneously generated or for individuals who did not respond to the prompt to think aloud. For example, in the event that a teacher did not respond to the general prompt "tell me what you are thinking," the interview would allow the researcher to offer probes that might elicit a more codable response. (See Appendix G Interview.) Not all participants completed the interview due to time constraints (see Section on Limitations in Study pp. 272) The probe questions focused on identification of learning objectives within the tasks in the games; assessment of whether or not learning outcomes would be achieved; and requests for evaluation of variables such as quality of feedback, graphics, and overall instructional merit.

Most participants provided enough data for the experimenter to code. Those who were reticent remained so during the interview.

What is evident from examining the responses generated during the interview is that most participants, both teachers and novices, were able to identify what they believed to be the intended learning outcomes, although novices tended to regard tasks not saliently mathematical, such as tasks directed as developing spatial ability, as lacking in mathematical content.

> "…I guess uh seems pretty alright. I don't see...basically I don't really see anything mathematical about it. Maybe I'm missing something…"
> 6N Appendix F p.339

Summary

A qualitative analysis of the data suggests that Teachers and Novices are invoking different schemas in their approach to the software evaluation task.

Novices' representation of the problem appears to differ from Teachers as evidenced by their beliefs that exposure leads to learning, so software that is highly engaging will lead to longer exposure, resulting in desired learning outcomes. For them, the problem of software evaluation appears, at least in part, to be one of evaluating the software for its motivational aspects.

Their database of pedagogical information is limited so they do not automatically recognize implicit learning objectives in tasks--thus, they expend their cognitive energies focusing on instances where a task does not appear to them to be relevant to the stated learning objective, that is, they do not identify tasks that are not obviously mathematical and focus on where the math is in that particular task.

Novices focus on most salient aspects of the problem; they get so involved in the tasks present in the software (as evidenced by their attention to solving the math problems) that they often neglect to determine whether or not the task is instructionally effective. They lack the content knowledge, pedagogical content knowledge, and pedagogical knowledge (Berliner, 1991) necessary to effectively evaluate instruction.

Teachers possess knowledge of what makes for effective instruction, and, as evidenced by their responses, have represented the problem differently from novices. They do not believe that exposure leads to learning; they focus on pedagogical concerns such as feedback, rather than directing their concentration primarily to features that might engage the player of the game.

It appears that while Teachers are able to employ the strategies they have for evaluating traditional instructional modalities to the electronic platform, they do lack the expertise with which Educational Technologists perform. Educational Technologists have a comfort level with educational software which affords them a richer schema for software evaluation.

CHAPTER VII

Discussion

Overview

If one believes that the materials used in the classroom have a significant impact on instruction, then the ability to evaluate those materials for instructional effectiveness should be of utmost concern to educators. Computer technology is going to be a part of the instructional process for teachers and students at least in the foreseeable future. Thus, the ability to evaluate educational software is a skill that teachers already need, and will continue to require.

The purpose of this study was not to evaluate the teachers' ability to effectively evaluate software--it was only to explore their problem solving approaches in this area. This distinction is important, for while evaluating teachers' ability to effectively judge software is an extremely important question, it is not the question under investigation in this study. In order to evaluate teachers' abilities, one would need a much larger sample, a more accurate means of assessing the teachers' pedagogical knowledge, and most importantly, an established metric for given software packages on particular variables. Thus, the focus of this study is simply whether a participant considered a particular variable important enough to mention, and if they had formed an opinion regarding the instructional merit of the software. It is not the focus of the study to determine whether or not those opinions are valid. In other words, if a participant claims that the feedback is excellent for a given reason, the researcher can note that the participant focused on feedback for this reason, an occurrence which allows the researcher to understand the problem solving strategies employed by this participant. Whether the feedback was actually good--that is, whether the participant was correct in his or her assessment is a matter for another study.

This study is concerned with problem solving strategies among teachers and novices in unfamiliar domains--the research question is whether the unfamiliar domain of technology impacts the problem solving abilities of teachers. The Educational Technologists provide a baseline for responses that can be expected from individuals who possess training and experience in both teaching and technology domains.

Conclusions

The expert-novice paradigm is a useful model in understanding the relationship between experience in a given domain and the ability to solve problems within that domain. The introduction of educational software into classrooms requires that teachers be proficient not only in using the software, but in evaluating it for educational merit, much in the same way they evaluate texts, lessons, and other materials at their disposal. However, unlike textbooks, educational software is operated in a computer environment which may, for many teachers, be unfamiliar enough to diminish their ability to solve this problem using their expert schemas; instead forcing them to rely upon the strategies typically employed by novices. Analysis of the findings of this study illuminate some of the issues that surround the question of whether educational software creates an environment unfamiliar enough to teachers to diminish their expertise in problem solving within their domains.

Differences Between Groups Across the Major Focus Categories

Teachers generated more statements than novices, a fact not surprising considering that teachers are used to talking; talking is a requirement for their profession. Novices were college students who may not have been as comfortable with talking about the software with a stranger. Furthermore, novices were participating in the experiment as a requirement for credit in their introductory psychology course. Although they were told that their participation in the experiment would in no way influence their grade and that there was no correct response, some novices expressed concern over whether or not they were performing correctly, and it is possible that concern over performance inhibited their generation of utterances. It is possible that novices, perhaps less experienced with evaluation tasks, concentrated on just being able to make utterances, as directed by the think aloud procedure, and thus, generated more statements in the Other category for want of statements to make.

It can be inferred that Teachers generated a greater percentage of Pedagogical statements than Novices because they are applying the knowledge that they have about learning and instruction to the software evaluation task. Teachers are also generated a greater percentage of Technical statements, a finding which at first may seem contradictory with expectations for teachers based on the initial hypothesis that it would be Novices who generated the greater percentage of Technical statements. However, closer examination of the distribution of responses on the Technical variable cluster reveals that Teachers are not necessarily focusing on the more salient variables of the Technical category, such as audio or graphics, but rather, are concerned with hardware and programming issues because they recognize that overly difficult games are likely to create obstacles to learning.

Technical Category Examined

Familiarity With Computers. Teachers generated a greater percentage of statements pertaining to Programming (TI) and Hardware (TK) than Novices. Although the mean number of hours spent using the computer for Teachers was almost twice that of Novices (3.71 vs. 6.07,) the mean age for Teachers was 42 compared with the mean age for novices at 19.57. Thus, for many of the Novices, personal computers are almost as familiar as the VCR or microwave, and while Teachers who use computers may be spending time on them, computers may not be quite as "second nature" to them as they are to the novices who grew up with them. It is likely that Novices would not be as attentive to hardware-oriented issues.

Further, since many of the Teachers did not grow up in the information age and are less familiar with computer technology than Novices, it follows that they may have been more preoccupied with instructions on how to play, or ease of play, as well as comments about using the computer itself, including mouse, keyboard and other hardware related concerns. There was an absence of Hardware comments from Educational Technologists (See Appendix E p. 282). This may be because of their high degree of experience using computer technology. It is interesting to note that Educational Technologists generated a greater percentage of Programming or Software-related (TI) statements than did teachers or novices. It can be inferred that having knowledge of software enabled them to make a greater number of comments about the interface.

Familiarity With the Stimuli. It is not surprising that the majority of technical statements are generated under the cluster TG (Game-Related) for both Teachers and Novices. Since both Teachers and Novices are seeing the games for the first time, it can be expected that many of their comments may focus on their experience with the game itself. Even the Educational Technologists had a high percentage of game related questions for Outnumbered (See Appendix H p. 410). However, Educational Technologists learned the games more quickly than

Teachers or Novices, and if they couldn't learn a game, they abandoned it, recognizing that overly complicated games probably meant more limited educational games.

Content Category Examined

Novices exceeded Teachers in the percentage of Content variables they generated. This finding suggests that Content variables may be perceived by novices as the most salient variables in this task.

Only Math Workshop showed significant differences between the groups on the Content Variable Clusters, with the differences greatest for Problem Solving (CP). For Math Workshop Novices generated a higher percentage of statements than teachers in this category. A possible explanation is that Novices were more willing to play the game for the sake of the game than Teachers, thus generating more statements on the solving of puzzles. For example, many Teachers, and the Educational Technologists, encountered complex or confusing games and quickly exited, noting that their task was one of evaluation, not actual participation in the solving of a puzzle. Additionally, Novices may not have been as comfortable with the math as is evidenced by some of their statements. Finally, Math Workshop has many more instances of non-salient math experiences, that is, tasks which stress spatial ability, but do not appear to be "mathematics" in the sense of arithmetic. Thus, Novices might be likely to take note of this, and indeed, many commented on the lack of "math" in a number of the tasks.

The lack of Accuracy of Information (CA) statements for either package can probably be attributed to the fact that the software is highly commercial and the domain, elementary mathematics, is so readily accessible that the question of accuracy would be taken as a given by participants.

Pedagogical Category Examined

It would appear that teachers are responding differently, but not exceptionally so, from Novices based the number of responses generated across the variable clusters within the Pedagogical category. There are many possible explanations; some lie in the integrity of the experimental design, (discussed in the section on Limitations of the Study) others in the problem solving processes of the groups.

Teachers generated not more than six percent more Pedagogical statements than Novices for either software package. Although Teachers generated a higher percentage of Pedagogical statements than Novices, the difference between groups on the percentage of statements generated within the variable clusters was not significant. A qualitative analysis of the data suggests that Teachers have different schemas for instructional effectiveness than Novices; that is, they do not equate exposure to material with guaranteed learning outcomes, and they focus on variables associated with instructional effectiveness such as feedback. However, the quantitative data do not reveal as strong a focus on pedagogical issues as on technical issues.

The Educational Technologists generated 19 percent of their statements in the Pedagogical category for Math Workshop compared with 18 percent for the Teachers (not much difference) but for Outnumbered, Educational Technologists generated 25 percent of their statements in the Pedagogical category compared with 12 percent generated by the Teachers. It seems that with a more complicated software package, such as Outnumbered, Teachers are not as able to fully access their schemas for evaluating effective instruction as Educational Technologists, perhaps because they are distracted by the complexities of the environment. It appears that Teachers are operating with a somewhat diminished set of problem solving strategies when in the less familiar domain of technology, for while they do generate more pedagogical statements than Novices, they are not generating that many more.

Of course, the sample is small, so it is difficult to make definitive statements regarding these results.

Explanation Statements Across and Within Groups

Across Groups. It was hypothesized that Teachers would generate a greater percentage of Explanation Based on Experience or Personal Belief (EE), and Explanation Based On Theory or Principle (EP) statements than Novices. Teachers generated a slightly higher percentage of EE statements Novices for Math Workshop, (11.5 percent of their total responses as Explanation Based on Experience or Personal Belief statements compared with 9.2 for the Novices) and for Outnumbered (9.6 compared with 4.4.) Only .9 percent of Teachers' explanation statements were generated as Explanation Based On Theory or Principle compared with .3 percent for Novices for Math Workshop, and only .1 percent for Outnumbered (Novices generated no EP statements for Outnumbered.

Educational Technologists generated 13 percent of their Explanation responses based on Experience or Personal Belief for Math Workshop and 14 percent for Outnumbered. They generated three percent of their

Explanation statements Based on Theory or Principle for Math Workshop and two percent for Outnumbered.

This suggests that teachers are not drawing on their experiential or theoretical knowledge, on the set of rules and concepts they have for effective instruction to the degree that might have been expected when they engage in the problem of evaluating software. Alternatively, they are not articulating such thought processes via their think-aloud responses.

Within Technical. It appears that the Description attribute is where the greatest number of responses are generated for both groups. Description statements are perhaps the most easily generated, with the exception of the Other category, in that they do not require any critical assessment. A possible explanation is that the Technical category, from which the majority of responses were generated, lends itself to Description statements. For example, it contains a category of responses that refer to a participant's experience with the game (TG). TG accounted for the majority of responses in the Technical category. Any utterance that involves a participant's playing of the game, for example, "I'm going to walk through this door," or "He shot me!" would be coded TGD, a factor which probably contributed to the large proportion of Description statements across the categories.

Within Content. There are no significant differences between the groups for Explanation statements generated in this category. It can be inferred that Novices and Teachers are invoking similar schemas when evaluating Content-oriented issues. There is a complete absence of Explanations Based on Theory or Principle for Novices in Math Workshop, and only one response for Teachers in this category.

The Content category is comprised of variables that do not require substantiation to the degree that the Pedagogical category does. Attributes like what problem solving approach to take, for example, do not require explanations to the degree that an observation might.

Within Pedagogical. The Pedagogical category reflects the greatest number of Explanation Based on Experience or Personal Belief (EE) and Explanation Based on Pedagogical Theory or Principle (EP) statements. While there is no significant difference between the groups, there are relatively few Description (D) statements for this category. Apparently, both Teachers and Novices felt obligated to substantiate their reactions to the Pedagogical variables. It can be inferred that the participants are comfortable merely making statements in the Technical and Content categories

-- this may be because of the nature of these categories. A statement about how enjoyable the game is, or whether or not something could be considered "Math," does not require evidence in the way that a statement about feedback might.

Summary of Conclusion. The findings of this study suggest that there are distinctions between the way Teachers and Novices approach the software evaluation problem, however these distinctions may not be as dramatic as might be expected and may not lie in the areas that might be predicted. For example, while Teachers generated a greater percentage of Pedagogical statements than Novices, for both software packages, the degree to which they exceed Novices is relatively minor, when compared with the degree to which they exceed Novices in generating Technical statements. Moreover, Teachers are generating relatively few statements Based on Experience or Pedagogical Theory. To what degree can these results be connected to the computer environment? Examination of the Educational Technologists' responses suggests that there is a relationship. Table 17 presents a comparison of the Teachers and Educational Technologists for clarity of discussion.

It appears that the more complex software package, Outnumbered, elicits greater differences in the percentage of Pedagogical statements generated and Explanations Based on Experience or Theory. This suggests that the more complex the software, the greater the impact on the problem solving skills Teachers possess when evaluating software.

Table 17

Comparison of Percentages of Statements Generated by Educational Technologists and Teachers and Novices

	T		C		P		EE		EP	
	MW	ON	MW	ON	MW	ON	MW	ON	MW	ON
E. T.*	44	46	17	10	19	25	13	17	3	2
Teachers	52	56	10	12	18	11	12	10	.9	1
Novices	37	39	19	17	11	8	9	4	.3	

Note: E. T. = Educational Technologists MW = Math Workshop; ON = Outnumbered. T = Technical; C = Content; P = Pedagogical; EE = Explanation Based on Experience or Personal Belief; EP = Explanation Based on Theory or Principle. All percentages have been rounded to the nearest tenth.

*n = 2 for Educational Technologists were a group of two; n = 14 for Teachers and Novices.

Limitations in the Study

Some limitations inherent in the study itself with respect to the design must be mentioned. The length of the experiment is perhaps the most unwieldy variable that was dealt with by both the experimenter and the participants. Because both the Novices and Teachers often did not have expertise in the domain of educational software, sometimes as long as 10 to 15 minutes per game was spent in the instruction phase. To increase generalizability, two games were selected as stimuli, but the use of two games resulted in a longer experimental procedure, and, it is hypothesized, fatigue on the part of the participants that may have lead them to take a cursory approach to completing the evaluation and a reluctance to complete the interview.

Many participants did not fully or even partially complete the interview, claiming they had no more time to spare. This is most true for the teachers, who were often devoting a lunch hour or after school session to the experiment, and had plans immediately after.

The participant pool poses issues for experimental integrity. Participants were self-selected for both the Novices' and the Teachers' groups. The pool of participants from which Novices were drawn was from a large city university introductory psychology course. Participants were required by the university to participate in two studies as fulfillment of course requirements for the introductory psychology course. They were permitted to select the studies of their choice from all the available studies run over the course of the semester. Some participants may have approached the assignment with a negative attitude, wishing only to complete it as quickly as possible to receive the credit, and may not have been conscientious participants. The fact that participants selected this study may also contribute to subject bias; they probably had some interest in technology, particularly computer games. Participants who enjoy computer games may have had a predisposition to enjoy these games, reducing their ability to critique them negatively.

Additionally, at least half of the novices did not speak English as their first language. While being a native speaker was not a requirement for participation in the study, some confusion may have resulted from a misunderstanding of the language. Many of the Novices initially believed the experimenter's use of the term "instruction" referred to directions on how to play the game, not "education," as the experimenter intended.

The experimenter never modeled the evaluation task during the "think-aloud" training in the interest of not influencing the participants in their focus. However, it is possible that the Novices, as a result, never accurately conceptualized the task. More authentic modeling of the experimental task may have provided the Novices, who had no formal background in education, with a better sense of what was expected, or what the task involved.

Finally, Novices did not have command of the language of teaching, for example, they had no knowledge of terms like feedback. However, this is not to say that they had no understanding of this, or appreciation of its role in a learning situation.

Experimental integrity issues exist for the Teachers group as well. Most Teachers were extremely reluctant to participate once they learned that the study could require as much as an hour and a half, or more, of their time. Many required payment for their participation. As with the Novices, reluctance to participate may have resulted in a cursory approach to the task.

Additionally, it is not known if these participants would have performed effectively on any evaluation task; they may not have been any more successful at evaluating a text book for educational merit. Thus, it is not possible to fully determine the extent to which the unfamiliar environment of technology impacted their evaluation strategies since it is not known how these participants approach an evaluation task in more familiar territory.

Several participants refused to allow the experimenter to thoroughly explain how to play the games, insisting that either a) they preferred to evaluate the games without knowledge of how to play as part of their evaluation would include how intuitive the game was or; b) that they believed, after a very brief overview of the game, that they understood how to play. This approach nearly always resulted in a large (relative to those who were willing to have a thorough explanation) number of questions for the experimenter on how to play, and, it can be inferred, a greater focus on the playing of the game itself.

The sample size is not large enough to safely generalize across populations. The Educational Technologist participant pool was very small and selected as a baseline of what true expertise in both the teaching and technology domain would yield but again, it was too small to be generalizable.

It is not possible to discern, from the analysis, whether contributions to the data are being generated by a few participants or by many. That is, it is not possible to tell whether say, 200 statements in a given category come from one participant or 13, unless the distribution of responses is examined for individual participants (see Table H6 in Appendix H p. 394.)

Role of the Researcher.

The researcher was the sole individual responsible for collecting data. It might have increased the reliability of the experiment to have another individual doing the data collection. Intimate knowledge of the software may have influenced the experimenter to too quickly rush through the explanation of how to play. Additionally, the threat to reliability exists in the fact that the experimenter has knowledge of the hypothesis and expectations of how the data should reflect that hypothesis. Bias exists as a result, and can be seen in the willingness of the experimenter to draw inferences from the participants' remarks in real time, as opposed to pressing them for explanations where statements and comments are actually ambiguous.

Moreover, the researcher was one of only two coders.

Intimate knowledge of the research question, coupled with an expectation of the results the data should suggest, may have influenced the coding schemas of the researcher. The coding process was also extremely lengthy, and the researcher and coder both experienced fatigue which may have compromised the accuracy with which they coded the responses.

Strengths of Study

Liabilities aside, this study provides a basis for future research in this area. By replicating the study with a larger sample, or redefining the research design slightly, researchers may be able to answer some important applied research questions that will help better prepare educational institutions in their teacher training programs.

Suggestions for Further Research

One possible way to redefine the study without losing the research goals might be to increase the participant pool in terms of number, but decrease the amount of time required by participants for participation in the experiment. The latter could be accomplished by selecting a large number of software packages to be evaluated, but requiring that each participant evaluate just one package. This would increase the generalizability of the findings as a larger number of software packages would be used, and the subject matter would not necessarily have to be limited to mathematics, thus allowing elementary school teachers who teach a variety of subject areas to evaluate software in more of these areas. This change would necessitate increasing the subject pool. However, since most participants reported a reluctance to participate based on the length of the experiment, the shorter time commitment might encourage greater participation.

Another option is to allow teachers to take a copy of the software and the small tape recorder home to perform the experiment at their convenience. An advantage to this is that participants are more comfortable with their own computers, are probably more willing to focus on the experiment if they are not pressed for time, and thus, may be more willing to participate. A training period with the experimenter would still be required however, and of course, much of the control is compromised are probably more willing to focus on the experiment if they are not pressed for time, and thus, may be more willing to participate

While some refining may increase the reliability of the experiment, and certainly reduce experiment, and certainly reduce the practical constraints, the experiment as it exists still allows researchers a window into the cognitive operations of Novices and Teachers evaluating software. This affords researchers some valuable data from which they can draw inferences about the state of education for teachers with respect to the programs for educational technology.

APPENDIX A

SUMMARY OF SOFTWARE PACKAGES

Math Workshop

This math software package has seven components that users can select from. A brief discussion of the components follows:

Bowling—Essentially a drill during which players select a math application, division, subtraction, etc., and a level of difficulty. For each example correctly answered, a gorilla will knock down a pin when he bowls at the end of the series of examples.

Puzzle Patterns-Players see an in-tact electronic jig-saw puzzle and must then recreate it when the computer scrambles the pieces.

Pattern Windows-Players assemble a picture-less jig-saw puzzle. When the pieces are assembled to form a rectangle, the picture appears.

Tracing Game—Players drag lines to place them on the outline of a graphic.

Rockets-Players take turns with computer (Polly) or a friend in sending off no more than a given number of rockets. The player who has rockets at the end of the series of turns is the winner.

Music-Players use their knowledge of fractions to divide and then reassemble a bar to meet the requirements of the computer. For example, they may be asked to divide the bar into thirds using implements that will only allow the bar to be cut into sixths. Players must then assembled the sixths into thirds.

Art-Players select from a collection of grids and then choose colors from a pallet to place within the grid lines to create patterns.

Outnumbered

This math software package is a game in which players try to score points and increase their rank from a trainee to a senior through successful completion of drill and problems. Players use a mouse or cursor arrow to move throughout two floors of a television studio, searching for the head of the studio who is hiding in one of five rooms. The complete set of code reveals the location of the head of the studio.

In order to obtain pieces of code, the players must zap a wandering robot-like television set with their remote controls. If they zap it, it will present them with a series of math examples. Completion of the series of ten examples yields a piece of code.

Players get energy to "zap" the television set by going into the rooms in the studio and clicking on appropriate places in the rooms to reveal a math word problem. Players increase their energy levels by solving these problems. Their energy levels diminish by bumping into the robot TV set while wandering throughout the studio, or by getting "zapped" by this TV set.

Players must locate the head of the studio within a certain time limit, or they do not win the game.

APPENDIX B

INSTRUCTIONS FOR PARTICIPANTS

Instructions

You have been selected to participate in a study that examines how people evaluate educational software packages. The computer has been loaded with two educational software packages for you to evaluate in the subject area of elementary school mathematics. You will have up to 30 minutes to explore each package but you may end your session with each package at any time before then, or elect to terminate the experiment at any point in the session.

Your evaluation should be based on whether or not you believe that the software is <u>instructionally effective</u>. You may want to comment on features or aspects of the software that you believe have an impact on how <u>instructionally effective</u> the software package is.

While you are examining the software packages, you will be commenting or "thinking aloud." This means that you must voice any thoughts that you have while you are using the software. The experimenter may prompt you to voice your thoughts by asking "what are you thinking and why" or to further explain comments that you make.

Your computer monitor will be videotaped and your comments will be audio taped.

Before you begin evaluating the software, you will be asked to fill out a brief survey requesting biographical data. Following the survey, you will have the opportunity to practice using the computer by playing a game of solitaire. At this time, you will also have the opportunity to practice "thinking aloud."

When you feel comfortable with the computer and the think aloud process, the experimenter will load the first software package and demonstrate how it works. You may then begin exploring the software package. You may explore any or all of the features in the software package.

Feel free at any time to request help from the experimenter if you need help using the software or the computer.

When you have completed your evaluation of the first software package, the experimenter will then load the second software package and demonstrate how it works. You will follow the same procedure for the second software package, thinking aloud as you explore it. Once again, your computer monitor will be videotaped and your comments will be audio taped.
Following your session with the second software package, the experimenter will ask you your opinion of specific features in each software package, and some questions regarding your opinion of the instructional effectiveness of the packages. Your response will be audio taped.

Finally, you will be asked to fill out a brief software evaluation questionnaire for each package.

Thank you for participating in this study.

APPENDIX C

CONSENT FORM

Dear Participant,

I am a graduate student at the City University of New York and am currently working on my dissertation. My research question involves examining the differences in the way expert teachers and novices (those without teaching experience or background) evaluate educational software.

If you agree to participate in the study, you will be asked to evaluate two educational software packages for elementary school mathematics instruction.

You will be asked to explore both packages for a period of time not exceeding 30 minutes per package.

During your exploration, the computer monitor on which you work will be video-taped and your voice will be audio-taped. Your voice will be captured on the video-tape as well.

You may end your participation in the study at any time.

Your anominity will be protected. Your identity will not be revealed.

These tapes will be reviewed by researchers or research staff and will be used only for professional purposes.

The tapes will be shown only to professional colleagues for purposes of explaining the research.

In order to qualify to participate as an expert teacher in this study, you must meet the following criteria:

a) You must have a minimum of five years teaching experience;
b) You must have excellent evaluations from a principal or supervisor;
c) You must be comfortable using a personal computer; and
d) You must not have taken a professional course in educational technology.

In order to qualify to participate an a novice in this study, you must meet the following criteria:

a) You must not have been employed in a teaching capacity in an educational or professional setting;
b) You must not have a college or graduate degree in education, or have taken professional courses in education, or in educational technology; and
c) You must feel comfortable using a personal computer.

Your signature below indicates that you have read this consent form, that you meet the required criteria for participation, and that you agree to participate in the study.

Signature of Participant ____________________________

Date _________

APPENDIX D

SAMPLE SOFTWARE EVALUATION FORM

(Identical form is used for Outnumbered; Actual form fits on one page and choices for items fit on one line.)

SOFTWARE EVALUATION SURVEY

Name______________________________

Please complete the following survey. For items 1-15, please circle the response that best reflects your opinion of the software. For items 16 and 17, please fill in the blanks provided.

MATH WORKSHOP

1. **The information presented in the software is accurate.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

2. **Learning objectives are clearly stated in the software.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

3. **The software presents concepts effectively.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

4. **The software uses meaningful or appropriate examples and problems to reinforce concepts.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

5. **Students would be motivated to use the software.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

6. **The software contains useful feedback on student's responses to questions or problems.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

7. **The software assesses where student has weaknesses.**
   Strongly Agree Somewhat Agree Somewhat Disagree
   Strongly Disagree No Opinion

8. **The software contains a means of assessment of**

student's progress.
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

9. **The software provides an opportunity for students to develop and use problem solving skills.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

10. **The software is largely "drill and practice," that is, students practice specific skills, for example, their multiplication tables.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

11. **The software could be effectively used by students of varying ability levels.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

12. **The pacing of the software seems appropriate. (It doesn't move too slowly or too quickly.)**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

13. **The graphics in the software enhance its overall effectiveness.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

14. **The audio in the software enhances its overall effectiveness.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

15. **The software is relatively easy to learn to use.**
Strongly Agree Somewhat Agree Somewhat Disagree Strongly Disagree No Opinion

16. **Please indicate the grade(s) for which you would recommend this software.** __________

17. **On a scale of 1 to 10, with 10 being exceptionally good, and 1 being exceptionally poor, please rate the overall instructional effectiveness of the software.**

____

APPENDIX E

BIOGRAPHICAL INFORMATION FROM TEACHERS

Software Selection Survey
The purpose of this survey is identify some of the factors that influence teachers when they select a software package for use in their classrooms. Your participation in this project is greatly appreciated.

1. Name____

2. Age_______ 3. Gender_______

4. Racial/Ethnic Background (Check one)

Caucasion _____
Hispanic _____
Asian _____
African American _____
Other _______________ (indicate)

5. Highest level of education (Check one)

High School or Equivalent _____
Some College _____
Associate's Degree _____
Bachelor's Degree _____
Some Graduate School _____
Graduate Degree _____
Other ______________________________
(Indicate)

6. How many years have you been teaching?

____

7. What is the name and address of the school at which you are currently teaching?________________________

8. What grade are you currently teaching? ________

9. What grade(s) have you taught? _________________

10. What subject area (math, language arts, etc.) is your main interest? _____________________________

11. Approximately how many hours per week do you use a computer with your class? _______

12. Do you have a computer(s) in your classroom?_______

    How many? ______

13. What educational software packages are you familiar with?____ ___

14. Do you own a home computer?_________

15. Approximately how many hours per week do you use a computer for personal use?_________

16. What software packages do you use? ________________________

17. Have you ever seen the programs

    Outnumbered!? _________

    Math Workshop _________

BIOGRAPHICAL INFORMATION FROM NOVICES

Software Selection Survey
The purpose of this survey is identify some of the factors that influence people when they select an educational software package. Your participation in this project is greatly appreciated.

1. Name____

2. Age_______ 3. Gender_______

4. Racial/Ethnic Background (Check one)

Caucasion	_____
Hispanic	_____
Asian	_____
African American	____
Other _______________	(indicate)

5. Highest level of education (Check one)

High School or Equivalent	_____
Some College	_____
Associate's Degree	_____
Bachelor's Degree	_____
Some Graduate School	_____
Graduate Degree	_____
Other ______________________________ (Indicate)	

6. What is the name and address of the school at which you are currently attending classes?

____

7. What is your major program of study?

____

8. Do you own a home computer?________

9. Approximately how many hours per week do you use a computer for personal, academic, and/or business use?________

10. What educational software packages are you familiar with?____

11. What software packages (word processors, spreadsheets, etc.)do you use?

____

12. Have you ever seen the programs

Outnumbered!? ________

Math Workshop ________

APPENDIX F

PROTOCOL TRANSCRIPTIONS

Participant 23 T
Age: 39
Gender: F
Ethnicity: Other (Jamaican American)
Education: Masters
Years Teaching: 20
Grade: 4-6
Area of Interest: Math
Own Computer: Yes
Hours Per Week on Computer: 6
Educational Software: Yes
Computers In Class 2
Hours on Computer in Class: 3

MATH WORKSHOP

MUSIC GAME
Ok. I want to get out of here.
TGD

I don't like this because uh, like this doesn't tell me what to do.
TID
PARTICIPANT EVALUATES THE EASE OF PLAY. THIS PARTICIPANT REFUSED TO LET THE EXPERIMENTER TEACH HER THE GAME. HER RATIONALE IN REFUSING INSTRUCTION WAS THAT SHE WANTED TO BE ABLE TO EVALUATE HOW INTUITIVE THE GAME WAS, AN IMPORTANT ISSUE FOR HER, SINCE SHE HAD A LARGE CLASS AND LIMITED TIME AND RESOURCES. THE GIST OF HER CRITICAL ANALYSIS REVOLVES AROUND THE EASE WITH WHICH CHILDREN WILL BE ABLE TO LEARN THE GAME-SHE DOES NOT FOCUS ON WHETHER, IF THEY DO IN FACT LEARN THE GAME, THEY WILL BE LEARNING SOMETHING AS A RESULT OF PLAYING IT.

I can tell you a little about how to play this one and then you can decide

No but that's what I'm trying to say. There are no instructions.
TID

Right, you can actually put the bar there and then hit do it.

So they're learning fractions.
COD

But you have to know how to do it.
ON
PARTICIPANT REMARKS THAT WHILE THE CONTENT IS THERE, A USER MUST BE ABLE TO PLAY THE GAME TO BENEFIT FROM IT.

So it plays a song and tells you you did it.
PAD

You can get another question by clicking on one of the green
bars.

OK.
OF
I think you can cut it in two pieces if you want

Oh, OK that's the same thing I don't like in the game.
TID

What's the blue for?
TGR

You can glue some of the pieces together, so you want to ... cause the object is you want to be able to drag them down there.

Oh, oh oh.
OF

Ok Oh.
OF

They give you a certain number of pieces.

Right.
OF

Oh he's anxious today
ON
Do you have any thoughts or comments about it?

Yeah I, one thing I don't like about it, is that it doesn't give, it doesn't give directions. You just have

to feel your way around. You have to feel your way around it. It doesn't tell you, do this, do that. For my kids, they would have to have instruction.
TIEE
PARTICIPANT IS EVALUATING THE SOFTWARE. SHE IS FOCUSING ON THE LACK OF INSTRUCTIONS ON LINE.

They would...I haven't done this...they would love this, its just that I'd have to be there. You know in the beginning to give them the instruction and explain it to them. Probably the teacher tells them how to do it and they can help the others.
PFEE
HERE THE PARTICIPANT IDENTIFIES WAYS THE SOFTWARE COULD BE USED IN THE CLASSROOM, BUT POINTS OUT SHE WOULD HAVE TO BE INVOLVED IN SETTING THE CHILDREN UP. SHE IS USING HER OWN EXPERIENCE WITH HER CLASS TO DETERMINE WHAT THEY WOULD BE ABLE TO DO INDEPENDENTLY.

OK.
OF

You can try other parts if you like.

Yeah, I'm gonna exit this one.
TGD

See you.
OF

TRACING GAME

Now you just have to drag those lines so you can make that pumpkin.

Oh no!
OF

The purple button will rotate the particular graphic that you're clicking on.

I am horrible at art!
OD

This is rotated?
TGR

If you click that purple, yeah it will rotate.

Good, that's what I wanted.
TGD

OK. This is good for the little kids, first grade, first and second grade I think would enjoy this. Some of my big ones would like it.
PIEE

Uhm, not that way. No. A little bit.
TGD

Oh Ok. let's see. This piece needs to be turned around.
CPD

You see, uh, like what Polly told me, you have to do this, you have to do that, there should be some written instructions, otherwise I would have to be with them every step of the way to get them to do this. And they would probably even be better at it then I am.
TIEE
AGAIN, PARTICIPANT IS EVALUATING THE QUALITY OF INSTRUCTIONS IN THE GAME.

Actually, you can drag them right on to the pumpkin.

Oh,
OF

Like right

there.
OF

Yeah, I hear you.
OF

Uh, I have to remember to double click.
TK

are you thinking anything?

Me?
OF

Any comments you can make?

Yes, as I said before, it would be difficult for the kids to do this without any instruction,
TIEN

It's user friendly,
TIEN

but I would have to explain to them what would be done. Which is no
problem, its just that..what I would do is just show them how they're supposed to do it, and then they could show the others.
PID
PARTICIPANT REITERATES HOW SHE WOULD USE THE SOFTWARE WITH HER CLASS IN TERMS OF THE PRACTICAL CONSTRAINTS OF TEACHING THEM THE GAME.

Oh boy, almost there. This one I need to....
TGD

You see like if you didn't tell me what I could use to turn it, I wouldn't know.
TID

But it is interesting.
TGEN

Do you want to try any of those?

Let me see what this one says and then, I would choose another game from there.
OD
You can play as many as you like.

ROCKETS

Ok so the object of this game...

(selects help)

Oh this one is a little bit better than the one before cause at least she explains, you know the kids may not be sure what they heard but they can you know, rethink.
TIEE

Ok.
OF

You can click on the rockets to send them up. And the object is, you want to be the last person to send up a rocket.

Yeah but for
OF

You just click the rocket. And they'll shoot up.

(help selected)

So you want to click on Polly's exit. you see, she's got the little red sign in the hand.

So what the numbers there for?
TGR

You can change the number of rockets that are on the launch pad, so if you want fewer or a greater number, you could change that. And you could change the number of rockets you're sending.

You can send up to three. And then when you want her to go you click on her.

And then...
TGR

She's done. Well what happened? I still have this one.
TGR

You won, cause you have the last rocket.

Oh.
OF

Oh.
OF

Uhm, I didn't get here what the numbers were.
TGR

You know we had 11 rockets on the pad,

right
OF

and if you click, say 12,

Oh.
OF

It'll give you 12.

I've got it.
OF

Well this this, I like this one because she uhm, she's interacting with you when its your turn, when its her turn, how many she's gonna send.
TIEN

I only have one left.
TGD

I did that, I did this one. OK. Oh no, I did this one already.
OD

This is similar to the other one.

Oh the drawing? Oh I don't want to do that.
OD

OUTNUMBERED

There you go.

Oh he's back?
OF

You keep going? Or you go in here again?
TGR

OK you should go in another room. If you can hit the space bar to shoot him, you'll get a real problem. You are out of energy so you also might want to look for this little place which looks like an antenna to stand on to get energy.

You don't have enough energy to kill him.

But you know what kids will do? They'll just look to kill, they won't look for the problem.
PAEE

Ok where does he go?
TGR

You're making me crazy. By the time we get over here

we'll be dead meat. (Unintelligible)
TGD

Why doesn't he walk?
TGR

Come on, I need a problem.
TGD

We might have to restart.

I'm afraid he doesn't want to go.
TGD

Sometimes if you drag, if you drag him a little bit.

Here. Now where is he supposed to stop?
TGR

Elevator...if you want.

You're dead.
TGD

Open up. It doesn't open. OK.
TGD

Oh (unintelligible) he shoots too much. I won't get a problem.
TGD

Too slow. Too much action.
TGEN

Want to give it another try?

I won't get this one.
TGD
PARTICIPANT BELIEVES SHE WILL NOT BE ABLE TO PLAY THIS GAME SUCCESSFULLY
AND QUITS.

PROBES
MATHWORKSHOP

1)
For one, mainly skills, drills, and practice. Not a lot of problem solving, word problems I didn't see many of

that. Mainly drills and practice.
COD

2) bowling
Oh that one was fun.
TGEN

That had a lot of drill and practice,
PID

multiplication, subtraction addition, and so on.
COD

4)
Fun for one, precision, you have to develop a space thing, order.
COD

5)
That was much better. Cause at least the kids could pick a whole, and divide it into different sections and see like for example, how many sixths are in twelvths, and so that was good.
PIEE

6)
I didn't learn much from that. I didn't see the objective.
PID

9) Polly
One good thing about her, she gave the instructions, what to do, when to start,
TIEE

you could play with her if you wanted to play with an imaginary friend.
PFD

10)
Part of it that I liked was that if you made a mistake, it would show you. Like when I did the estimation. some of them that I missed it would leave it there, so if I have a good memory, I would remember, you know, (unintelligible)
PAEE

11)

They weren't hard to use. There wasn't much technical difficulties with that.
TIEN

12)
It would have been good if that while Polly was giving instructions the kids could read along, you know that could strengthen more of reading problems, that could strengthen their reading skills.
PID

Except in the bowling, it was very good, but it wouldn't tell you like specifically what you did wrong, why your answer if incorrect.
PAEE

13)
As reinforcement I would buy, and I would use it as reinforcement. Yeah with other materials, like after teaching my lesson, the kids who are finished, additional practice, or those who work slow, put them (unintelligible).
PID

OUTNUMBERED

1)
I didn't like that one. That one I really didn't get into.
ON

2)
Well, for testing for the kids, they know, you know that some of those tests are timed and they have to finish in a certain amount of time
COD

but if you don't understand what you're doing, you're going to fail
TGEE

3)
I couldn't move, I don't know.
TGD

The kids probably they would do better because they're good at video games and so on I'm slow at that. I didn't know when he was going to shoot, or when to move him

along, whatever, but I think the kids would do better at that than I did.
TGD

5)
No feedback.
PAD

8)
I would, but I would have to become very familiar with it so I could teach the kids how to use it.
TID

Participant 17 T
Age: 55
Gender: M
Ethnicity: White
Education: Masters
Years Teaching: 33
Grade: 4-6
Area of Interest: Language Arts
Own Computer: Yes
Hours Per Week on Computer: 20
Educational Software: No
Computers In Class 1
Hours on Computer in Class: 5

OUTNUMBERED

What grade level are we talking about?
PIR
ONE CAN INFER FROM THIS STATEMENT THAT SUBJECT IS INTERESTED IN AGE-APPROPRIATENESS OF THE SOFTWARE.

Supposedly for ages 7-10,

Ok
OF

That's what its recommended for.

I just wanna get all the math problems or it will be embarrassing. OK
OD

That music is driving me crazy, I'll tell you that much. It's steadily non stop the same thing consistently-- its enough to cause somebody to get a little looney.
TAEE
SUBJECT COMMENTS ON THE MUSIC; HE CLAIMS NOT TO HAVE MUCH EXPERIENCE WITH EDUCATIONAL SOFTWARE-PERHAPS THAT IS WHY HE IS SO ATTENTIVE TO WHAT IS ESSENTIALLY UNREMARKABLE BACKGROUND MUSIC.

Now do I need to move it?
TGR

Yes, you can hold the mouse button down

hold the mouse button down and try to drag it.
TKD

Sometimes if he's in the middle of shooting he won't walk.

I should have gone like that, right?
TKR

Right, you've got the hang of it,

Yeah but I didn't. OK Just hit the mouse button.
TKD

I got the hang of it that much, you have to hit this, is that what it is?
TGR

Yes, actually the space bar.

This is the space bar? Or this is?
TKR

Yeah, right there

Yeah, oh...you can see that I didn't know the space bar Ok
TKD
THE SUBJECT IS NOT THAT COMFORTABLE USING THE COMPUTER IN THIS CAPACITY. HE DID NOT KNOW WHICH KEY WAS THE SPACE BAR, AND IS CLEARLY NOT THAT FACILE WITH COMPUTER TECHNOLOGY. THIS MAY HAVE SOME IMPACT ON HOW HE EVALUATES THE SOFTWARE; HE MAY BE FORCED TO CONCENTRATE TOO MUCH ON NAVIGATING IN A COMPUTER ENVIRONMENT TO GIVE HIS FULL ATTENTION TO THE EVALUATION TASK. THE GIST OF HIS CRITICAL ANALYSIS IS BASED ON THE SOFTWARE'S REQUIREMENTS FOR TECHNICAL ACUMEN-HE IS NOT AS FOCUSED ON THE QUALITY OF THE INSTRUCTION IN THE GAME, ASSUMING ONE HAD THE TECHNICAL EXPERTISE TO PLAY IT.

Alright now, I should just go to the right or the left? and hope that somehow or other a door is going to arrive?
TDR

Whichever choice you like. there you go, now he's walking uh oh..if you put the mouse on the other side of him you might be able to hit him or maybe...looks like we're out of energy go in the room and try to answer problems

Ok
OF

and that will help you. When you go to the energy thing it'll work

Now where do I go to answer problems?
TGR

You can click on anything. You never know where they're hidden

They might be hidden there, they might be by the presents

OK, alright
OF

There's definitely a question
TGD

There's a question someplace
TGD

yes
OF

here we are
TGD
(mouthing the question)

21 dollars difference
CPD

do we do this one enter
TGD

OK ya got it

Wow
OF

Same damn music...darn music OK
TAD

Ok I have to go to leave the room and go to try to

find..go in the other rooms
TGD

now how why did he sneak back in the room again
TGR
HERE WE SEE MORE EVIDENCE OF SUBJECT'S LACK OF EXPERIENCE WITH THE MOUSE. HE HAS SOME DIFFICULTY CONTROLLING HIS CHARACTER AND IS UNSURE WHY. THIS MAY DISTRACT HIM TO SOME EXTENT FROM THE SOFTWARE EVALUATION TASK.

oh, just depends on...where the mouse is

he can't stay in that room right?
TGR

No cause we already did the question in there

so we can look for other rooms
TGD

OK now normally we could have gotten him only since we're out of energy

oh I see yeah so that out of energy thing is a problem
TGD

because
OF

oh I should have gone in the room
TGD

its not working because I don't have energy? Is that what its telling me?
TGR
SUBJECT SEEMS TO BE SOMEWHAT PREOCCUPIED WITH THE GAME. HE IS NOT COMMENTING ON HOW THE COMPLEXITY OF THE GAME MAY IMPACT THE LEARNING OUTCOMES ALTHOUGH HE TOUCHES UPON THIS LATER.

right

cause you don't have energy

so I should go back in this room again?
TGR

or another room. Will you look and see if there's any problems

it's the same damn room
TGD

that's the same room...there's 4 other rooms

ok
OF

can I go in the elevator
TGR

sure, you can go in the elevator

what do you think of that
ON

now what do I do?
TGR

get on the elevator

the elevator doesn't do nothing for me, just to find another room
TGD

right

I don't understand...he doesn't have energy, I thought I hit that right away?
TGR

You're out of energy, see, so you're gun is empty

uh huh
OF

and when you answer more word problems..uh oh we're out of time, so we didn't win

ok its' very frustrating and it runs out of energy so fast
TGEE

you can give it a couple of minutes to try one more time and then we'll change games

ok
OF

now we've got energy so now if you see you can actually shoot him

ok
OF

almost

I really think that a youngster, however, would do better
OD

ah you got em

a youngster would probably do better in that sense cause they're used to these kind of games so that uh, cause they play these things in arcades and so forth so much more and I don't
ON
SUBJECT IDENTIFIES HIS OWN WEAKNESS WITH COMPUTER GAMES, MENTIONING THAT YOUNGSTERS WOULD OUTPERFORM HIM. HOWEVER, HE IS NOT EVALUATING THE SOFTWARE - JUST HIS OWN PERFORMANCE. PRESUMABLY, THE UNFAMLIAR ENVIRONMENT IS DISTRACTING HIM FROM THE EVALUATION TASK.

oh you enter, see that's what I'm saying, I'm not used to this, so I'm certain that from that standpoint I'm just going very slow cause I'm not used to it that's a problem here
TIEE

...2 plus 8
CPD

ok
OF

can you comment on any instructional aspects

well, you said what age group?
PIR

7 to 10

you got a piece of the puzzle

obviously very very basic but for seven its not
PIEN
HERE SUBJECT EVALUATES THE AGE APPROPRIATENESS OF THE SOFTWARE.

well you can adjust the levels of the math

why am I not moving this?
TGR

oh you can adjust
TGD

Oh Ok, now how do you get him? Oh, Ok? How do I know? How do I know that?
TGR

Well we only answered one word problem .. we'd have to answer more word problems

I don't see why that would not work.
TGR
HIS CONFUSION OVER THE GAME SEEMED TO DISTRACT HIM FROM DELVING FURTHER INTO THE ISSUES OF AGE APPROPRIATENESS.

Out of energy. But maybe you can answer a word problem

This energy thing is a pain in the neck
TGEN

Yeah. Maybe if you answer a word problem.

OK.
OF

But again, I'm spending more time on this stuff on the mechanics than the educational value of the whole thing
PIEE
SUBJECT FINALLY IDENTIFIES AN IMPORTANT PEDAGOGICAL ISSUE-THE COMPLEXITY OF THE GAME MAY BE INTERFERING WITH ITS ABILITY TO TEACH. MANY TEACHERS ADDRESSED THIS ISSUE.

because I'm so nimble and fast its hurting the fact that I'm not spending any time
OD

Now I gotta go try to find a
ON

Yes

puzzle
TGD

oh that's good at least there's a change of song
TAEN
(mumbles while solving)

Well the fact that I'm not particularly sophisticated with games
OD

and kids but its not something that would make me be judgmental against it
OD

but the only thing is that I think that a lot of times kids
(unintelligible) the mechanics of it and a person like me who's not all that inclined that way
TKD

like right now why isn't this thing moving
TGR

Yeah, you have to just drag it in front of it

Well I am, but
TGD

and then hold it down, yeah there is goes

so that penalizes me and uh there's too much penalty on that versus penalty for not knowing answers, if you know what I mean
PAEE
SUBJECT REMARKS THAT THE EMPHASIS SEEMS TO BE ON THE GAME, AND NOT THE INSTRUCTION, IN THAT THERE IS GREATER PENALTY FOR NOT BE FACILE WITH THE COMPUTER THAN FOR HAVING MATH DEFICITS.

and it could be very frustrating for somebody that's they're like me and they don't know the way to walk.
TGEE

Would you like me to start the other game?

no let me finish it...
OF

so I mean there's enough fascination to see whether I can figure this out for a change
PAEE

now this is really frustrating to me....here we go
TGEE

(mumbles)
now I should try to find the puzzle
TGD
(mumbles while solving)

the excitement of the music makes you happy that's for sure
TAEE

well, now I'm suppose to leave the room
TGD

yeah

I can't even figure out how to leave the room...OK
TGD

there's another room

sometimes there might be an extra puzzle in there

yeah
OF

(reads the problem)

what is that number?
TID

looks like an 82 and that is...

that's 88. oh, I didn't see the 88 for a second
TAD
(more mumbling as he solves)

shouldn't the minus go first?
CAR

it was 82 it was reduced by 88 so its minus 6 but this is 6 minus ..is that alright? you do it that way?
CPR

I thought I did this right
CPD

oh maybe that's 68
CPD

I can't read very well
TID
(mumbling)

oh 80
CPD

oh its 80
CPD

again, again I would have a problem with that (unintelligible)
ON

equals 2 equals
CPD

how do I get a 2 I'm trying to get a 2 here
TKD
SUBJECT IS HAVING DIFFICULTY SOLVING THE PROBLEM, PRESUMABLY DUE TO THE TECHNICAL DEMANDS AND CONSTRAINTS. HE DISCUSSES THIS IN THE FOLLOWING STATEMENTS.

any other comments you can think of?

no I just think that because I do have a tendency not to be all that comfortable with (unintelligible) it's tremendously frustrating
TKEE

and therefore, even if you know the answers you find yourself now after a while even though got the hang of it but if you have a youngster that has those kind of manipulative problems I would think that uh he's being

penalized
ON

and it would detract (unintelligible) and then after a while
ON

this energy thing would be annoying
TGEN

MATH WORKSHOP

The fact that she is a Latino is good because all of these kind of waspy oriented or even Black but never somebody who is (unintelligible) Latino Latina
CSEP

You can go here here here here here or here

OK
OF

BOWLING GAME
you can just hit go

Right. I see.
OF

and then I just give the answers?
TGR

just click on the answer that you think is correct

OK, there we are
TGD

now it's exciting when you know what you're doing
TGEE

(mumbling)

it's reasonably challenging
CDEN

and also I think what I like about this is it's uh even if you're--you

don't have to be terribly dexterous to do it
TKD
SUBJECT NOTES HOW FACILE ONE MUST BE WITH THE MOUSE--SOMETHING WHICH IS OF GREAT IMPORTANCE TO HIM. HE MENTIONS THIS BEFORE PEDAGOGICAL CONCERNS.

anyway
OF

that's good
OF

what do I do?
TGR

I'll do another one
TGD

when you're doing it right and you know what you're doing you kind of stay with it.
PAEE

that's why I'm saying the other one could be frustrating for kids that are not dexterous or don't have the hang of it
ON

and the teacher with 25 or 30 in the class is spending a lot of time helping each kid do this
PFD
CLASSROOM CONSTRAINT ISSUE

and that's frustrating all around
PFEN

well we might as well get out of here, right? where's the exit?
TGR

there's the exit

right
OF

and he bowls the way I would
OD

he uses contemporary language too which is good, I mean,

the language of the kids, chill out and stuff
PIEP
NOTES AFFECTIVE ELEMENTS OF THE SOFTWARE

now the other games are where?
TGR

you can click on the television or the butterfly

choose your picture

PUZZLE GAME

now I'm suppose to put these things on?
TGR

yeah (unintelligible)

and then?
TGR

and then it's a question of timing is that the idea?
TGR

no, there's no timing

oh, there is no timing, this is more of a psychological test
COD

this could be very embarrassing
OD

so you just have to make sure you put them all in
TGD

whoops
OF

(mumbling)

and then you wonder why kids killing teachers
OD

I get frustrated with this stuff
OD

There's no upside down or anything like that

TGR

Yes, the purple button rotates the pieces for you

It will? I can click this thing?
TGR

Just by clicking that purple button

don't worry, in about 2 hours I'll have it all finished for you
OD

now
OF

da da
OF

That's the end of it.
TGD

You got it. Do you have any comments?

So that's just basically to determine how you see shapes
COD

and there isn't that much
CIEN

Well I can see a tremendous amount of frustration for a youngster having trouble manipulating a situation
TKEE
AGAIN DISCUSSION OF MANIPULATION OF HARDWARE

Oh I don't want this...now see, even myself, I'm reacting against it,oh God, please don't give me another shape I'll start to get rather hostile
TGEE

Right, so the only ones without shapes at this point

hmm hmmm
OF

the fraction machine or the rockets

but uhm, if each kid is doing it and one kid is starting

to be a loser there's a potential discipline problem and he won't be happy
PAEP
SUBJECT DRAWS ON HIS PEDAGOGICAL KNOWLEDGE IN THIS STATEMENT.

MUSIC GAME

now what am I doing?
TGR

It wants you to make one half bar and one quarter bar in two quarter bars. and you need to drag the blue bar

oh oh I see
OF

and just click the cut it--there's only opportunity to cut it into two. just click and let go

and now you can click on that number 2 where it says cut it

(unintelligible)

right there?
TGR

no

oh I'm sorry
OF

how many people, how many youngsters would you have doing this?
PFR
CLASSROOM MANAGEMENT QUESTION

oh that would be up to you as the teacher You could have them work in buddies, or alone or in groups

uh huh
OF

and now what am I suppose to do?
TGR

you want to drag down to that yellow box

yeah so you just drag the (unintelligible) all the way down uhm, drag this piece into there
OF

so you have half the puzzle solved (unintelligible) how to get that piece in order

this one?
TGR

yeah

into what now?
TGR

put it up again

right
OF

and and you know click two

and yeah
OF

just say do it. It would would work.

Yeah, Do it.
OF

and now you can drag the quarters down

and that way you can show (unintelligible)
ON

yeah

I get the message.
OD

and do you did it

right
OF

and how do you determine what age group? 7 to 10.

PIR

that's just what the documentation on the box suggests but you as the teacher may make a deci--may feel differently

would everybody, well basically,
ON

ok
OF

alrighty

for this game now that we just played
OD

PROBES
1)
uhm, fractional equivalents, in concrete ways, measurement skills
COD

2)
the bowling thing was, well basically there were there were various--well division, multiplication, basic concepts of addition, subtraction multiplication, division
COD

and its painless
PAEN

5)
well I thought that was the fractional equivalency
COD

8)
well those, and I remember even as a youngster, not, I remember getting 55 in art you wouldn't be surprised seeing how I played the game uhm,
OD

and so if you're skilled in one way, because of your artistic problems it would be tremendously frustrating and conceivably could have a negative impact on your enjoyment of it, of the math puzzle because you're having such trouble with it.

ON

9)
her being Latina was positive, it was a positive stereotype, identity for those Latinas
CSD

10)
Well it may be repetitive but I think that you know, it's exciting, it's interesting, it's fascinating and challenging,
PAEE

but at the same time, the point which too much, too much is dependent upon your skillfulness and your manipulation of materials, takes away sometimes from the fact that the excitement of doing well is overshadowed by the inability to manipulate the instrument. that can later on, I suppose, be a problem.
TKEE

11)
the only technical would be to make sure because especially if you have a number of youngsters playing you can't possibly individualize with each kid
PFEE

and conceivably, the manipulative skills would take away from the academic excitement
TKEE

12)
no question it's interesting, its fascinating, its a charicature, cartoon type of figures, and kids, probably more than people my age would would really relate much quicker to this they enjoy games they play arcades all the time so probably they wouldn't not be as frustrated as I'm saying, they're relating to something that's very positive and exciting for them, and they would rather those little cartoon figures than the teachers that are there in the room.
PAEE

13)
I would, but I think what I always liked was uh, I had a

lot volunteer adults coming into the building and um, therefore, with either a paraprofessional coming into the classroom, the aid in the classroom, or the volunteers, one on one, that would be great. I would be hesitant to use it in a large setup.
PFEE

OUTNUMBERED

1)
well, that yeah, Can I see the objectives in that? Well again, I think that...just being frustrated by the energizer was a problem, where the heck is everything supposed to be, on the other hand, for those who are skilled at that, they can figure out the answers and thats good.
TGEE

2)
some of us, we work...yeah that's a problem for some of us because that timing situation creates certain tension` for a number of people, there's a tension factor involved, which um, after a while, you would just give up, oh I'm not gonna bother doing this anymore
PAEE

3)
yeah, I had good impression of him, I just thought it was annoying, I didn't think it added at all to the whole dimension of learning and I thought that in some ways but again that's me, I just didn't uh, it was an annoying it was an added frustration that didn't add much to the learning situation and was annoying to me.
PAEE

5)
yeah, when the music gets exciting and the monkey tells you how good you are (the first game)
I didn't think it was particularly exciting
PAEN

7)
Again I would, that you can move at your own pace, I suppose, so and with a miminum of teacher involvement, you could supposedly take one of these games by yourself and play them once you've learned and as long as you're not using a timing technique, and you've started to learn

the mechanics by doing it yourself,
PFEE

its keeping you busy, but its the kind of busy work that's (unintelligible) I love the busywork that teachers give kids...
COD

8)
Oh I would use it in the classroom because different you use strokes for different folks, certain kids will be really turned on by it
PIEP

Participant 1N
Age: 18
Gender: F
Ethnicity: Hispanic
Major: Social Science
Own Computer: No
Hours Per Week on Computer: 2
Educational Software: No

OUTNUMBERED
Alright.
OF

Do they use this to, uhm, in school help to help kids.
ON
Some schools use it. (unintelligible)

I think it does help because there's a lot of math problems
COD
SUBJECT BELIEVES THAT PRESENCE OF PROBLEMS LEADS TO LEARNING. THIS IS A PREMISE SHARED BY MANY OF THE NOVICES.

and you know, they have, I don't know.
ON

It's really educational, I think.
PIEN
SUBJECT CANNOT EXPLAIN WHY SHE THINKS THIS IS EDUCATIONAL.

Any comments?

What age group? Is this for elementary school?
PIR

Yes.

Oh.
OF

Maybe you can go in one of the rooms.

And then there are word problems in there. and you can click on different things to find the word problems.

I can click on what?
TGD

You can click on the

on anything
TGD

Yeah and some of them will have word problems. Not all of them, some. It just depends on where you click. Maybe the map might be.

The map?
TGD

Maybe the orange screen.
TGD

No.
OF

Did we ever go in this room? No we didn't.
OD

So get out right.
OD

There should be a problem in here. Let's try that poster

Which one.
TGR

That one.

Let's see if there

Nah, I clicked on that.
TGR

OK let's get out then. We'll try to find another room.

The arrow going that way is up?
TKR

Yeah. (unintelligble)

And sometimes you run low on energy

Oh that's why
OF

And if you go into

in a room
OF

yeah there's a little square that gives you energy, I think on another floor.

I don't have any energy.
TGD

Yeah I think we're almost out of time

You can try it again if you like.

(restarted)

The arrow, the black arrow comes out when he goes?
TGR

Usually the arrow is connected with the mouse. The TV comes out by itself.

He's not moving.
TGD

OK keep him--oh
Ah, looks like you almost got him.

I'm out of energy again.
TGD

OK. See that green thing. You can get energy by standing in that.

OK let's go stand in the energizer.

Oh.

(finds a problem)
Now you tell me what you think...Tell me what you're

thinking.

Tell me what you think.

Oh, I yeah. I didn't read the question carefully.
CPD

This is really helpful for little kids because they should learn this, you know, right now, while they're in, uhm, junior high school, because they're going to need this math,
COEE
SUBJECT BELIEVES THAT THE PROGRAM IS HELPFUL BECAUSE IT COVERS MATH THAT STUDENTS WILL NEED, BUT AGAIN, SHE DOES NOT EVALUATE WHY IT IS HELPFUL...APPARENTLY IT IS SUFFICIENT THAT SUBJECT MATTER IS PRESENTED; SHE DOES NOT EVALUATE THE PRESENTATION.

because when you're coming into college, you know you have to take a placement test and a lot of people fail it where they forget their basic steps, their basic math that you have to know.
COEE

And now, you see, they're even gonna take that away, with CUNY, that they're gonna take away the remedial classes, so its better if you get prepared while you're young and you don't have to regret it.
OD

There's no questions here?
TGR

Usually one question per room.

Oh OK.
OF

Oh I guess you didn't have any energy.

He can't jump or anything?
TGR

Take the elevator.
TGD

OK.

Hmmm.
OF

Should be here. This one?
TGR

Ah there it is.

(silently reads problem)

Can you tell me what you're thinking?

Oh, I'm reading the question now, I'm sorry.
CPD
SUBJECT FINDS IT DIFFICULT TO CONCENTRATE WHEN READING THE QUESTION. THE MATH IS NOT AUTOMATIC FOR HER-LIKE NOVICES IN OTHER DOMAINS, SHE TOO FINDS IT COGNITIVELY TAXING TO EFFECTIVELY PARTICIPATE IN A NEW TASK WHILE TRYING TO CONCENTRATE ON ADDITIONAL TASKS THAT ARE NOT AUTOMATIC. IN THIS CASE, SOFTWARE EVALUATION IS UNFAMILIAR AND THE MATH IS NOT AUTOMATIC FOR HER.

I think this is a great way for uh children to learn how to
do word problems and things like that cause like for some kids there's usually an art that they don't know, if its to add, or multiply, or divide.
PIEE
SUBJECT IDENTIFIES OBJECTIVES IN THE SOFTWARE, BUT CANNOT EVALUATE THE SOFTWARE FOR INSTRUCTIONAL EFFECTIVENESS. THE SOFTWARE DOES NOT ACTUALLY PROVIDE INSTRUCTION ON HOW TO DO THE PROBLEMS BUT THIS SUBJECT SEEMS UNAWARE OF THAT.

(continued the game)

If you've had enough of the game you could quit too.

You don't have to.

Let me just do this one.
TGD
EXPERIMENTER REMINDS SUBJECT THAT SHE CAN QUIT IF SUBJECT HAS SEEN ENOUGH OF THE GAME. LIKE MANY OF THE NOVICES, THE SUBJECT APPEARS TO HAVE BECOME ENGROSSED IN THE GAME AND WANTS TO PLAY, YET THERE IS NO MENTION OF NEEDING MORE TIME TO BETTER EVALUATE THE GAME. IT APPEARS THAT THE SUBJECT SIMPLY WANTS TO FINISH THE GAME.

Take your time. I just didn't want you to think you had to.

You could look at the

The
ON
Yeah and you can check the decoder and see that you have a piece of code but none of the other pieces.

What do I do next?
TGR
You can go back to the station.

For most kids, if they actually see math as being fun, they start to realize hey, maybe I should start playing more and they get skills in math.
PAEE

Do you have to go in a specific order and maybe that's why you don't find the game?
TGR

Oh, no, you can go in any order that you like, but once you've done the room, then there won't be any more questions in there.

Its just that I never seem to find the question
TGD

Maybe try the other room

Uh.
OF

Yeah.
OF

OK.
OF

Nope.
OF

Nope.
OF
(game over)

Ok what do I do to get out now?
TGR

We can quit.
If you hit file, there should be a quit.

yeah, exit.
OF

MATH WORKSHOP

Tell me what you're thinking.

Oh. Uh.
OF

The way kids think, uh, is like that game, I don't know if you ever, if you're familiar with it, but I'm that little kids have to put, OK, there's many shapes and you have to put the finger on where the shape goes.
ON
SUBJECT COMPARES THIS GAME WITH ONE SHE HAS PLAYED BEFORE

I don't know that one.

In this one, they're all squares, but its similar because they have to know where it goes to make a familiar pattern, COD

yeah and its helpful because it makes them think and....and
PIEN
SUBJECT SUGGESTS THAT GAME IS HELPFUL BECAUSE IT MAKES KIDS
THINK. SHE DOES NOT ELABORATE ON WHY.

I think I got out by mistake.
OD

Oh that's no problem, no problem at all. Let me see where the mouse is. Thank you. Its time for a pair of glasses for me.

Can you tell me what you're thinking.

They're getting (unintelligible) it doesn't matter right?
ON

Oh no, they want you to have a certain pattern.

All of em?

TGR

Yeah,

Like the same animals go in the same
TGD

Like this is blue, with the crab fish crab. So you need to find something that does that.

Hmm.
OF

You know, some way to arrange those so that there's a pattern.

Oh, OK.
OF

I'm getting a headache.
OD

I don't think I'm getting this. I can't see what to do with these.
TGD

I would ... Let's see something else.
ON

I'll go bowling.
TGD

He's cute.
TAD

What do I do?
TGR

Just click on the type of arithmetic you want to practice and then click on go.

OK. Oh, these are easy.
CDEN

Do they get harder?
CDR

You can change the levels right?

TGR

Yes.

This is fun.
TGEN

He's cute. I like him. But
TAD

He is the only one bowling, right? Do the kids get to bowl?
TGR

No, not really.

Well, I think it would be more fun if like every time you finished the questions you could bowl. I don't know. Just something so you are more involved. But its fun.
TGEE

I'm gonna try a harder one. Just to see.
TGD

OK. OK. Yeah, these are a little harder.
CDEN

Maybe they could have different animals bowl. Not just the monkey, but or, kids could pick uhm which character is going to bowl and they could have like a lot, or not a lot, but a few different animals to pick. Or, I don't know.
TGEE

OK.
OF

OK.
OF

Hmm. What can I do here?
TGR

Oh, you just click on the color paint you want, and then click in the place you want to place the paint.

This is good. This is good to teach kids about patterns. They could copy patterns, or make up their own. It would be good for little kids to learn the colors.

PIEE

How can I erase?
TGR

Just click on the background color and then click in the place you want to erase.

OK. Oh, I see. OK. This is good. I think its fun
TGEN

I think older kids might like it.
ON

What can I do now?
TGR

Oh can you print what you make?
TGR

Yes.

OK.
OF
Did you want to see anymore?

I'm ready. I'm ... I've seen it. I have to go soon, so. I'm done.
OD

QUESTIONS
MATH WORKSHOP

1)
There's some math. With the bowling, they're learning to add, subtract, multiply, divide,
COD

and there's different levels so when kids get better they can move up.
PID

I didn't see any math in that one with the squares…
COD

that was just annoying for me,
OD

but maybe kids could be learning to think…you have to remember the picture.
PID

2)
The bowling was good. I think kids are going to learn from it. It makes it fun to practice.
PAEE

I think its good for little kids who have trouble with basic math.
PIEN

3)
I think it was good,
ON

but they should tell you more that you have to study the picture before
you start or something so you would know that.
TIEE

I don't see any math,
COD

but maybe it helps you to think.
COD

4)
I liked this game because kids could be creative, and they could learn about patterns and shapes
COD

and its easy to do.
TIEN

I think little kids would learn it right away,
PIEN

and they could print what they did,
TGD

and its fun
TGEN

9)

I don't know.. Nothing. I didn't really see her.
ON

10)
It was OK.
ON

I think some of the parts, they should give you some clue, some guidance, because its hard to tell what you're doing, if you're doing it right, if you have the right idea.
TIEE

11) I didn't think it was that - the bowling was good. That could teach kids math.
PIEN

12)
But the rest of it. I don't think it would really teach math. Maybe patterns, like that. I didn't see much math.
COD

It was OK. Like I said before, in some parts it would be helpful to have some guidance, to show you if you are doing it right.
PAEE

13)
I guess I would use it. Kids like to play on the computer so they would get some practice.
PAEE

OUTNUMBERED

1)
Well, I think they had a lot of help for basic math.
COD

I think word problems are very important because people have trouble with them and they need to get more practice than just working with the teacher.
PIEE

A lot of people don't understand this math, but maybe since this is fun, they would practice more.
PAEE

2)
I think the timing makes you work fast, and you have to think fast.
PIEE

Its good because that is usually how you have to do for math, for tests. I thought it was helpful
COEE

3)
He was OK. I think kids would like him. He was full of action.
TAEE

5)
I think it could help you more, like when you're in the rooms, they should give you a hint where the problem is, instead of having to click and not know if its there or not.
TIEE

6)
I think it was hard to use the mouse, but maybe kids would find it easy since they play a lot of computer games.
TKEE

7)
Everybody needs to learn this, so I think it is very helpful. Math comes up in your life and you really need to know it for almost anything you want to do.
COEE

If kids can learn it while they're young, then they will have an advantage later.
PAEE

8) I would use it.
ON

Participant 6N
Age: 18
Gender: M
Ethnicity: Hispanic
Major: Social Science
Own Computer: No
Hours Per Week on Computer: 0
Educational Software: No

OUTNUMBERED

OK tell me what you're thinking

OK basic word problems, basic arithmetic, I guess. And uh, oh, OK.. .
COD

Just go ahead and type the number.
TGD

Oh check me out. Oh wrong number, that's a careless mistake. Oh once again, I'm not thinking.
SUBJECT IS FOCUSING ON HIS OWN ABILITIES TO SOLVE THE MATH PROBLEMS BUT NOT THINKING OF THE SOFTWARE EVALUATION PROBLEM.

(solved it)
CPD

Its midnight already?
TGR

Yes, we have to start again, you ran out of time.

(restart)

Quick.
OF

Only thing about this, the music's kind of nerve wracking. I don't know how the kids will take it.
TAEN

Alright, here we go. I did that the first time, I don't know.
TGD

(reads problem)

I didn't even see the numbers on the other side.
CPD

That's good. Has a calculator on one side to help you out, that's good, in case there's any real big numbers or anything. That's good.
PIEE
SUBJECT DOESN'T MENTION THE EDUCATIONAL VALUE OF THE CALCULATOR--ONLY THAT IT CAN INSURE ONE DOESN'T GET STUMPED ON PROBLEMS THAT CAN'T BE SOLVED MENTALLY.

Oh.
OF

Nah, that can't be right.
CPD

Tell me what you're thinking.

Like I said before, what I like about this, it has the calculator along side, to help you out. In case you like, off the top of your head you really can't do the problems because of the numbers, which I'm experiencing right now.
PAEE

What's wrong?
CPR

Oh OK, My mistake.
CPD

(reads the problem)

I always make careless mistakes in math. OK.
OD
MORE FOCUS ON HIS OWN ABILITY.

Careless mistake again.
OD

Oh my mistake.
CPD

So simple.

CDEN

I don't know how to read.
OD

It's pretty good.
ON

Oh.
OF

The best thing is that you don't have that annoying music when you're doing the problem.
TAD

Oh OK.
OF

Have any thoughts?

Uhmm. Its just.. I was thinking what I was thinking before. If you review my work on the video tape you'll see that I shoulda used this when I was younger.
OD

It's still, its basic arithmetic.
COD

I make careless mistakes.
OD

Let's get outta here. One more time.
TGD

Go to the various rooms.
TGD

OK.
OF

Lets see.
OF

(reading and mumbling about the problem)

I don't understand this. Thirty.
CPD

OK. Guess I'm getting closer to the guy now.
TGD

You can always check in the decoder and see.

What I do? With these?
TGR

Ok you need to get the secret code pieces, and you get them by zapping the TV.

OK.
OF

So you need those as well.

I haven't done that. I've been running away from him.
TGD

Next time you see him....you want to hit him with the spacebar.

I already went this way.
TGD

Its kind of fun. It gives kids a sense of adventure while they're playing. It will keep them into the game, wanting to come back and play more.
PAEE

You got me.
OF

If you want to get him, you have to touch the spacebar.

Why am I not able to get him?
TGR

Oh, OK.
OF

This game's fun.
TGEN

But now if you take a look at your decoder, now you have two pieces...Try to match up.

So...
OF

It's likely to be...

The equipment room.
TGD

Or the newsroom. We don't know about that.

Oh, I didn't go there. So I can't really.
TGD

If I make a decision I would lose right?
TGR
SUBJECT IS INTERESTED IN SUCCEEDING IN THE GAME.

You could take a guess.

I'll see what happens.
TGD

I don't know, lets go for it. See what happens.
TGD

This is a fun game.
TGEN

So would that be the end of the game?
TGR

Yes, or you could play again if you wanted.

I like it. Its educational and fun. The whole thing for kids you gotta keep it fun for them, I feel.
PAEE

MATH WORKSHOP

BOWLING

Ok
OF

so the object is to get all ten pins?
TGR

just tell me what you're thinking

there should be something like the monkey should be doing. Like bowling instead of just standing there.
TIEE

He should be knocking the pins down. I don't know if you could do that though.
TIEE

You can't do that.
ON

OK so its levels?
PIR

Yes

Oh OK
OF

tell me what you're thinking.

Uhm, its pretty good so far.
ON

Gotta see how the levels are. Level one seems pretty basic.
CDD
SUBJECT SHOWS SOME INTEREST IN LEVELS OF DIFFICULTY.

Its cute.
ON

This is for elementary level?
PIR
Yeah.

OK.
OF

take a different...
ON

I don't know...I'm not good at division.
OD

Uh, seems good so far.
ON

It should attract the kids attention because it has a lot of, like cartoons animated always catches kids attention. It should be good
PAEE

For elementary it seems pretty basic, but to help them learn uh, mathematic equations at the basics its pretty good.
PIEE

Lets see, I don't think I know this off the top of my head (error) knew it.
CPD

Think I should go back, cause I don't know all this.
OD

Well like I said before, it seems like it'll catch their attention
PAEN
SUBJECT IS FOCUSING ON AFFECTIVE QUALITIES.

The only thing is trying to get the amount of computers you need to get it out there.
PIEE
SUBJECT MENTIONS THE PRACTICAL CONSTRAINTS OF GETTING COMPUTERS IN SCHOOLS.

I guess the applause at the end will give them like a sense of
accomplishment.
PAEE

Oh, that's cute.
ON

A different level....
PIEN

This is actually kind of fun, I wouldn't mind having something like this when I was younger, it would have been fun to have.
TGEN

The only thing I see with this is it would be challenging for them but it will also be entertaining, they won't get bored with it.

PAEE

Hmm.
OF

Basically education should be fun, so this one is pretty good, I like it.
PAEE

Should I just go ahead and exit?
ODR
Sure. And you can try any of the others that you like.

ROCKET GAME

Uh, yes. This is like uh, sort of process of elimination, I guess.
COD

I'm not really sure.
CPD

(unintelligible)

I wanna play again.
TGD

You can choose a different number of rockets.

OK.
OF

Oh this is nice.
ON

And you can also change the number of rockets that are
launchable.

Oh really?
OF

Yes, there is says three...
Five
CPD

Can you tell me what you're thinking

Pretty much entertaining,
TGEN

but math oriented?
COR
SUBJECT DOESN'T SEE ANY MATH OBJECTIVES IN THIS GAME.

I don't understand it cause I keep losing.
TGD

But it seems fun. A fun game.
TGEN

Let's see another one.
TGD

The whole thing I like about these, these games is that its animated.
TAD

That should keep their attention, the kids attention. I know it worked
for me.
PAEE

PUZZLE PATTERNS GAME

Is this supposed to like uh, like count how many targets it misses? Something like that?
TGR

Well the object is you're supposed to be able to get it to make that picture.

Its OK. Its a game, I don't know, I feel it might bore the kids a bit, cause all your doing it trying to fit these things together. Could be fun but after a while it gets boring.....Its OK.
PAEE

Just finish off here....
TGD

And from here I go to....
TGD

And after I finish, I go?
TGR

You can exit if you like.

That's it. Oh OK. I thought you get a great round of applause or something.
TGD

(laughs)

PUZZLE GAME
Here's putting the pieces in together
TGD

Anything that you're thinking.

I guess...so far, animation wise I guess its kind of fun,
TAEN

trying to put these pictures together....I don't know.
TGEN

Oh.
OF

Ok.
OF

tell me what you're thinking

I guess uh seems pretty alright. I don't see...basically I don't really see anything mathematical about it. Maybe I'm missing something.
COD
SUBJECT DOESN'T SEE THE MATHEMATICAL RELEVENCE

This is not fitting.
CPD

I don't know what's going on here.
CPD

I don't know.
CPD

Its not that...its not interesting to me, not because that I'm not making the puzzle, cause that's why, that'll get me more into it. But I don't know. I don't really care for it too much.

TGEE

(exits game)

So far, I like the bowling game.
OD

TRACING GAME

This game seems like fun.
TGEN

Cause the kids have to work at putting images together. In a way which I don't think they would have thought of at first.
COD

Get outta here (mumbling to game)
OD

Can you comment on what you're thinking?

Some of the pictures here, like most of them, they have
ON

oh...what happened here.
TGR

Oh no problem, we can just click on that icon

Sorry about that.
OD

No problem

Uhm, Has pictures of things that I guess we all see, common, that we see a lot around us, so I guess, you know, it'll give them a nice sense of what's going on,
COD

which I do not have.
OD

Finally, I got it.
CPD

No I don't. Do I?
CPR

Let's see what other ones they have.
TGD

Everybody likes clowns. This is good.
TAEE

I like the changes, you know that you have to make, in order to get these things.
COEN

Yeah.
OF

Oh, At least this one gives you some kind of music.
TAD

(he won)

Ha ha.
OF

(exits game)

MUSIC GAME

Can you tell me what you're thinking.

So far, I'm just trying to see what I can do here. See what happens if I get this right or not.
TGD

OK.
OF

(won)

Gives you some sort of education on beats for music. Help you out with tempos. Its nice.
COEE

You can put those other blue bars in the middle while you're cutting that

Oh really
OF

Like you could put that there.

Oh really
OF

While you're cutting one cause it only wants you to cut one at a time.

You can only cut one at a time so you need to drag one of the boxes down.

Fill her up?
TGR

Uh Oh.
OF

Oh, well that's already that size.
CPD

How about I put everything back.
CPD

I don't know really that much about music so
OD

Put it back into one piece again
CPD

Let's see what we could do here.
CPD

I didn't mean to do that.
CPD

OK.
OF

Gluer says I can't glue.
TGD

Hmmm.
OF

OK.
OF

(solved it)

What I like about this is it gives you understanding what the different counts are for music.
COEN

You really, you really get to know if you really pay attention to the music at the end of the game..
PIEN

I think that's right... Oh wrong.
CPD

I think I did all
TGD

Participant 30 E
Age: 36
Gender: F
Ethnicity: White
Education: Masters
Years Teaching: 10
Grade: 4-6
Area of Interest: Technology
Own Computer: Yes
Hours Per Week on Computer: 20
Educational Software: Yes
Computers In Class 12
Hours on Computer in Class: 4

MATH WORKSHOP

BOWLING GAME

This is basically a basic drill and practice.
PID
SUBJECT QUICKLY IDENTIFIES THE EDUCATIONAL FOCUS OF THE SOFTWARE-IT IS DRILL AND PRACTICE.

let's see what happens if I make a mistake.
TGD
SHE IS INTERESTED IN SEEING THE FEEDBACK. NOTE THAT THE SUBJECT
REQUIRES NO HELP IN PLAYING THE GAME-SHE HAS ENOUGH EXPERIENE WITH EDUCATIONAL SOFTWARE TO HAVE A WELL FUNCTIONING SCHEMA FOR HOW TO NAVIGATE IN MOST GAME ENVIRONMENTS-THIS SCHEMA MAY FREE UP HER COGNITIVE RESOURCES FOR MORE ANALYTICAL EVALUATIVE TASKS.

They tell you right away that you made a mistake and give you the right answer.
PAD

Does it give you another chance...they don't give you another chance to figure it out, do they?
PAD
AGAIN, SUBJECT IS CONCERNED WITH FEEDBACK.

OK.
OF

I may go right on to the next.

TGD

How easy are the easy ones?
CDR
SHE WANTS TO KNOW HOW BASIC THE MOST BASIC PROBLEMS ARE.

Do they ever get to bowl? Or do they just get the pins?

TPR

they get the pins and then the monkey bowls, actually, you don't really get to.

So what are we working toward here?
TGR

You want to get all ten pins and that's a strike.

Unintelligible
ON

I have hard time with drill and practice because I think this is -- there's not a whole lot of thinking that has to go into it, there's not a whole lot of
PIEP
SUBJECT DESCRIBES THE ISSUES INVOLVED IN DRILL AND PRACTICE-SHE IS DRAWING ON HER PEDAGOGICAL KNOWLEDGE BASE.

I guess they have to do some mental math.
PIEE

But the whole idea is its fun to watch what's happening and the only reason they're doing it is to see the monkey do the bowling
PAEE

and uh, I don't know that it needs to be fun all the time so uh,
PAEE

as far as drill and practice, I guess it has its place you know to reinforce some skills
PIEE
AGAIN, SUBJECT DISCUSSED THE RATIONALE BEHIND DRILL AND PRACTICE.

but I know a lot of kids go in and they play it for the

game and lets see the monkey bowl and I don't even know if the math concepts make any difference. that's how I feel about the monkey bowling.
PAEE

So that's it. And they get a certificate if the monkey gets a strike.
PAD
SHE IS ONE OF THE FEW SUBJECTS WHO NOTICED THE PRINTABLE CERTIFICATE. AGAIN, HER EXPERIENCE WITH EDUCATIONAL SOFTWARE MAY HAVE PROVIDED HER WITH SCHEMAS THAT ENABLE HER TO KNOW A SOFTWARE PACKAGE RELATIVELY QUICKLY, WITHOUT MUCH INSTRUCTION.

Is there anything else here?
TGR

I know he's got one strike. So they can play this for a while.
TGD

yeah.

OK.
OF
ROCKET GAME

OK. I want to play this rocket game.
TGD

This is interesting because...you ever play the game Moncala?
ON

Moncala is a similar...its in a little tray and there are ten little pockets in each and you have you start with four little stones in each one and the idea is to clear your side of the tray before the other person. A little bit of a strategy and I watch my seven year old play it.
COD
SUBJECT IS COMFORTABLE ENOUGH WITH THE GAME TO COMPARE IT WITH OTHER GAMES THAT REINFORCE SIMILAR CONCEPTS.
We play it together and to watch her count and to try to figure out and figure the strategy...its similar to this and she's she's really good at it. OK.
COD

So I can launch up to five rockets.
TGD

And I have to click on each one?
TGR

Yes.

Ok now when its her turn?
TGR

You just click on her.

Oh she's gonna do em all and I lost. Go ahead...let me see you win.
TGD

Oh this is good because they really have to think.
PIEE

Oh she's not gonna let me do that
TGD

Ok so now what do you do Polly? Reset it?
TGR

It always take me a while to figure out the strategy.
OD

So this is good though. They really have to think a little bit. Its not the greatest I've seen but...
PIEE

they have to plan ahead. Plan you're strategy.
COD
SUBJECT IDENTIFIES THE LEARNING OBJECTIVE IN THE ROCKETS GAME.

I'm really bad at this stuff.
OD

Now how come she won't play right away? Is she waiting for me to
TGR

Click on her body.

This will definitely appeal to little boys.

CSD

I think this will definitely appeal to little boys,
CSD

yeah, there is definitely some problem solving that needs to be done, a little bit of strategy
PID

although its really not that difficult to beat her so again, they're gonna come back to this game for rocket value.
PAEE
THIS SUBJECT IS CONCERNED THAT THE GAME IS NOT CHALLENGING ENOUGH SO STUDENTS WILL PLAY THE GAME JUST FOR THE SPECIAL EFFECTS AND NOT REALLY GAIN MUCH FROM AN EDUCATIONAL STANDPOINT.

PUZZLE PATTERNS GAME

now this is the one with the picture
TGD

Yeah, when you choose one you have to remember what it looks like so you can recreate it.

Now is there a way to, it only turns in certain ways...
TIR

It can rotate in all four ways.

There's no in between?
TIR

No it just goes at 90 degree angles.

That's difficult.
TIEN

You can flip it, you know, two more times.

That's not right. I guess I didn't pay too much attention to the picture.
CPD

I like this though.
TGEN

My son would be really good at this. He can make pictures in his mind. Actually worked.
ON

I guess I hit the wrong one. Oh because I hit help.
TGD

Is there a way to make her leave?
TGR

Once you click on her stop sign then she goes away.

This is hard.
CDEN

Kind of challenging though.
CDEN

Once again though, I guess, as a time filler, which is not really where we want to be...
PIEE
SUBJECT SEES THE GAME AS A TIME FILLER. SHE FINDS IT LACKING IN ITS ABILITY TO MEET ANY SIGNIFICANT LEARNING OBJECTIVE.

that's just not right, is it? Ok.
CPR

Well, you have to really think and you have to remember.
PID

So many of these are to use a cliche, edutainment.
PIEN
SUBJECT IS ABLE TO MAKE A DISTINCTION BETWEEN WHAT SHE BELIEVES WILL RESULT IS LEARNING OUTCOMES AND WHAT I ESSENTIALLY NOTHING MORE THAN A GAME. THIS THINKING DEPARTS FROM THE NOVICE CONCEPTION THAT EXPOSURE LEADS TO LEARNING. ADDITIONALLY, IT SHOWS HER RELUCTANCE TO ACCEPT THAT SOMETHING TEACHES JUST BECAUSE IT CLAIMS TO BE EDUCATIONAL.

What else... This one I needed to try.
OD

Yeah, you can try any ones that you like and you don't have to try them all.

TRACING GAME

Snowman. This one is a little different, not quite as difficult, cause you can see it.
CDEE

This is good for little ones I guess who are just starting out
PPD

...using some fine motor skills to work the mouse...recognizing shapes and how they fit together.
PPD

OK.
OF

Oh that's cute.
TGEN

Ok. Entertaining.
TGEN

OUTNUMBERED

Ok So I drag him, and he got me
TGD

Right cause you need to hit him with the space bar.

So how can I hit him?
TGR

Well you need to be facing him.

Yeah, if you're facing him and you hit the space bar, then

You have to flip your guy.

How do you...Can you turn him around?
TGR

Yeah, just turn the mouse...just drag the mouse.

Well, alright.
OF

Oh this is a video game.

COEN

I would never buy this game for my classroom. I would never even buy it for my kids.
OD
SUBJECT HAS FORMED AN OPINION OF THE GAME EARLY ON.

Can I enter it from the key pad?
TIR

Yeah, either one.

Oh, and I did it wrong. I hit the wrong answer.
TGD

And the thing is, what I find very interesting there's a lot of talking and a lot of things to remember about how to run this, so it makes you think this is for older children, but the problems are on a first grade level.
PIEE
SUBJECT NOTES AN EDUCATIONAL INCONSISTENCY IN THE SOFTWARE-READING REQUIREMENTS DRAMATICALLY EXCEED MATH REQUIREMENTS IN TERMS OF ABILITY.
We'll go this way.
TGR

Why am I getting that little donut? Am I hitting the wrong button?
TIR
Its just where your mouse is.

Ok.
OF

Keep it in the main area. As you get closer to the bottom....

OK, so I want to go in here.
TGD

It won't let me go in?
TGR

(doing word problem)

So did we answer our one question?
TGR

So we can leave.
TGD

I'm no good at video games.
OD

Let's hit the elevator. He got me again.
TGR

Uh. Have you used kids as your subjects? Or does it all have to be
teachers?
ON

Teachers and undergraduates.

I just think the educational value of this is....
ON

We ran out of time.

PROBES
MATHWORKSHOP

1)
I would say, problem solving, I guess, some drill and practice, reinforcing your basic computational skills, but and some of the others, the smaller programs I guess, but overall you want to know....overall was to reinforce math concepts I guess, was the overall objective.
COD

Maybe in a small way in some of the smaller programs, but I would never use it in a classroom as a piece of software. Never.
PIEN

2)
OK. The bowling game--the objectives specifically I would guess would be reinforcing computational skills.
COD

Uhm, Uh I guess they would be met. The incentive is to get as many strikes as possible so they continue to drill and practice, repeat... repeat... Is it met? Maybe. In part. Possible. I can't give a great answer for that, I don't know.
PAEE

3)
I guess the objective would be shape recognition. Problem solving. How can uh, just solving a puzzle.
COD

I'm not being very eloquent here.
OD

I guess just being able to complete the puzzle and recreate the picture.
COD

Uhm, I think that's definintely for a lower level. A younger child. The picture is there.
PIEE

But like I said, I think that there is some value in fine motor coordination, having to flip things around and use the mouse uh, the objective, I guess it was met.
COD

6)
Rockets the objective was strategy I guess, how to beat your opponant by
COD

7)
The objective I guess is to solve the problem, without the benefit...its recreating the puzzle without the benefit of having the picture there,
COD
so I guess they really have to pay attention to detail, so its focusing on memory, and uh taking a picture in your mind,
COD
its not a very good objective
PIEN

...Uh and I think that would be a good one, I actually liked that one. It was a challenge for me.
ON

I think the objective was a good one but I don't think it was difficult enough. There weren't enough rockets that needed to be launched and you really didn't have to think that deeply, so uh, you know.

PPEE

9)
She wasn't too intrusive. I didn't really have an impression of her.
ON

10)
It was OK. The snowman, I enjoyed the colors being added and the animation, so I guess that would be incentive to do it again,
PAEE

uhm the bowling, though once again its pretty much standard in all those kinds of software games, it give them some positive reinforcement,
PAD

but are they going to go back to play the game because they want to learn more and enrich themselves or because they want to see the snowman color up and wave his scarf.
PAEE

11)
Uh....Let me think. I don't know. The graphics were nice.
TAEN
Actually it was easy to figure out. It was pretty simple, once you get the lay of the land
TIEN

uhm, you wouldn't need a whole lot of guidance to do this. If you were going to give this to a student, it could be very independent.
PFEE

Uh, but it still has that kind of a game quality.
ON

12)
Well, maybe as a filler to do a little bit of reinforcement, but I wouldn't rely on it to teach anything. Does it teach anything? I don't think so.
PIEN

13)
Wouldn't use in the classroom
PIEN

OUTNUMBERED
1)
The general objective is to solve some sort of a mystery with some strange space alien. That's the overall objective and interspersed in there I guess its supposed to reinforce some math skills.
COD

Did I think they were met? No. Absolutely not.
PAEN

Well, I guess as far as the mystery is concerned, there is a little bit of a plot going on and that's fine
COEN

but uh, the math part of it, as I said its very complicated, a lot of talking, and a lot to listen to and think about before they actually start the game.
PIEE

And then once they start it, the math problems that are given are so simple, it doesn't seem to match up at all. I don't know who would this be for.
PIEE

2)
timed
No except that forces them to work as quickly as possible which I guess in some ways has its merits but, I don't know, it just reinforces sloppy work and just getting through it for the purpose of the game
PIEE

3)
irritating, constantly in your face, and appearing at any time. I didn't like him.
TGEE

I couldn't shoot him.
TGD

5)
uh, there..the feedback was basically when you were given a problem, it said congratulations or the right answer which is good, which anybody could have gotten those answers so it wasn't terrific, it was good, it was

positive.
PAEE

6)
kind of fun using, kind of a virtual reality with the room that was interesting, but technically its more like a Nintendo video game and you know as nice as that it technically and as wowing, I know you said not, apart from educational, but I just think...
TPEE

7)
educationally I thought its reinforcing the whole Nintendo generation and I have a real hard time with that.
PIEN

I have a son who we constantly use Nintendo as a as a negative reinforcement basically, we just keep taking it away because its bad and they all do it they're so focused on it, and the game quality,
ON

I just didn't think it was a good game at all.
TGEE

8)
Never. Never.

Participant 29 E
Age: 43
Gender: M
Ethnicity: White
Education: Masters
Years Teaching: 10
Grade: 4-6
Area of Interest: Technology
Own Computer: Yes
Hours Per Week on Computer: 20
Educational Software: Yes
Computers In Class 2
Hours on Computer in Class: 2

MATH WORKSHOP

OK. OK. My first impression is that I like the opening screen. It was very interesting looking, very colorful and entertaining,
TAEE

but, but, they don't tell you what to do to get in, you have to assume or click on, you know, you would end up having to investigate to click on a bunch of things, and then finally get to the door, or get to the door first, to know that you have to get inside, and there's nothing that tells you, you know, welcome, come in so, that's my first problem,
TID

in that depending on the age that this is intended for, if this is uh, third grade, second grade, fourth grade, fourth grade they might pick up on it, but second and third graders, I don't think its a first grade game, second or third, I think that they might wait for something to happen, and they might not get it the first time around.
TIEN
SUBJECT IS ATTENTIVE TO AGE APPROPRIATENESS OF TASKS

OK so here and then
OF

a kid will click on anything.
OD
Great.
OF

PUZZLE GAME

My second complaint, somebody should be explaining to me what I should do. Or at least what kind of game this is. I've gotta spend time figuring it out. So now, I have to, as a kid, I have to make the leap that I'm gonna take all these pieces and copy the picture or put it onto the picture so let's see if that's what I'm supposed to do.
TID

OK
OF

So this is a --this is a logic matching part.
COD

and I guarantee, oh I see
ON

this is something that you're going to have to figure out that you're going to have to rotate the piece to get it in, same thing with this piece, ah this piece, some pieces you have to rotate to get in, so lets just get the ones that will go in straight without any rotation. Uh, will that go? that would be reversed, will this go? hey, let's see if any of these will go in without any additional help, ok so now I'm gonna wanna look at one of these--oh there it is OK so click on that, let's see, no that doesn't work so ah, there you go, so now I can put that Z here, and and I'm gonna click it--it doesn't go there, so I'll move it there, doesn't go there so I'll keep rotating it, til its fit, alright.
CPD

So this is really basically a matching game COD

and again, I have to find that rotation button, or be made aware that its there to figure out that its something I can use,
TID

I'm not sure that a kid will get it. You know, right away. There's a certain amount of explanation that needs to be done
TID

so I've gotten it

TGD

so the frog animates, and (unintelligible)
TAD

that's cute.
TAD

OK so now, Oh, that's nice, OK if the kid sees that, uh (unintelligible)
ON
(System crashed)

Well we found our first problem (laughing)
ON

(REBOOTS)

Now lets see what happens...Another kid is gonna want to click on a bunch of stuff, none of it...none of it responds that's the door, its a waste of time.
TIEN

Now another kid, oh, so now I've got to see if I'm still in there, there I am, OK, thanks, alright so now I've tried that one, I'll try this one now.
TGD

ROCKET GAME

Now if she does five rockets, I don't know if I'm supposed to set five rockets or it says number of rockets..11..one two three four five six seven eight so that gives you the total number of rockets that are available.
TGD

what happens if I click one of them?
TGR

That launched one rocket.
TGD

I don't understand what's happening because I thought if I clicked one it would launch five, so now you have to click them one at a time but it says launch up to five rockets.
TGD

You can click on four if you want. Then click on Polly to make it her
turn.

Now, but you see, it doesn't say that. So that's the problem.
TIEE

A teacher would have to explain it.

This stuff should be self-explanatory. Software should explain itself to the children.
TIEP
SUBJECT HAS A SCHEMA OF THE STRUCTURES THAT SHOULD BE PRESENT IN EDUCATIONAL SOFTWARE.

OK. And there's a mistake, I should have launched...she could launch now all of them and that'll be that.I mean she probably will.
CPD

Ah see. So if I had launched six, she could have only launched five and I would have won.
CPD

OK so let me try to my strategy again, hang on, reset.
CPD

Ok so, (unintelligible) six, so if I launch
CPD

I got it, I think.
CPD

So you've got strategy, so you've got to learn that you've got to do strategy and then you can also pick how many rockets.
COD

I think that so far my only criticism of this is that they don't explain it and you kind of have to grope around for a while and figure it out,
TIEE

and maybe if a teacher doesn't mind leaving the kid to do self-discovery.
PIEP

There's gotta be a frustration level, you know,
PAEN

and I think that I think that there should be some small framework to let you know what the to let you know what to do how do you operate the cause I don't think you should have to first click on something to have it explain to you how it works.
TIEE

Alright. Exit, Yep.
OF

Alright, lets try the bowling.
TGD
BOWLING GAME

Oh again, and here we are and the flashing thing says go,
TGD

so if I'm a kid I guess I'm gonna, I'm gonna do everything that's optional I'm gonna see, OK so now I'll go to the go thing and see what that does.
ON

OK 6 plus 6.
CPD

OK.
OF

So my second complaint, well my first complaint about this, and my second complaint also, this is the second one, is that if you get the answer wrong, if you get the answer right they give you a pin which is fine, if you get the answer wrong they give you the answer right away and I don't like that I'd like them to give you like two more tries to try again. I don't like that. I don't like them giving you the answer right away. So that's a problem.
PAEN
SUBJECT CRITIQUES THE FEEDBACK

Now three screen in a row where the answers been in the same box. The right answer's in the same box, I could just keep clicking the same box. I wonder if that's a pattern, I wonder if they put all the right answers in

the same box. That's interesting.
PID

OK So he got a strike. I want to see...
TGD

Oh that's not really clear. Because if you're really not familiar with that function, it sounds like explanation not estimation.
PIEE
SUBJECT IS ATTENTIVE TO POTENTIAL SOURCES OF EDUCATIONALLY BASED CONFUSION FOR STUDENTS.

OK.
OF

See what happens there.
TGD

One sixth...
CPD

So the answers aren't in the same box.
PID

And now they are... that's two in a row, that's
PID

Alright. I've seen it.
OD

I think that might be a little boring for them.
PAEN

Yeah thanks.
OF

Alright I'll see what goes on here.
TDG
MUSIC GAME

Alright so this thing shows up, I have no idea what I'm looking at, it says fanfare 2 one half bars I see a fraction, one over one, it says do it, cut it, glue it, meld it, I have no idea what to do
TGD

I can give you some help

Oh no no no that doesn't matter, I'm looking at this I'm a teacher and I'm evaluating it, what are my children going to think, or you know, I guess the only thing I'm going to think is Ok, do it. So that doesn't mean anything
TID

so let's try the little figure here
TGD

that doesn't help you, so now, let's see empty, OK so I'll go to the help button.
TDG

Ok.
OF

I don't like...it looks like it would be an interesting thing but so now, (mumbles) didn't do anything, alright
ON

so if our parts list is that we have two one half bars
CPD

...what's this?
CPR

One over one is a fraction so I don't want
CPD

No idea what they want me to do.
TID

You need to drag this purple bar up there, and then you can click on cut it, and you need to cut it into eighths, and once you've done that you go to do it to actually make it happen, so you click on the do it, and then you can make some of those one-eighth pieces into quarters and then whenever you have the right size piece,this is confusing, I'll tell you right now.
TIEN

OK

this is too confusing.
TIEN

I mean if I, you know. Oh I should go to glue it, do it, no, now I'm gluing everything back to that
TID

You can move some pieces back to that empty area so that you're left with fewer

In other words, you can drag the one eighth down to the little slot

Oh
OF

so you're left with just a few to glue together.

I I I can't see, you know, its totally convoluted. I mean, its not clear, its not simple, and its not self explanatory. I don't like it. I mean I think not that things should always be easy, but your figuring out how to play the game is an obstacle to learning what you're supposed to learn.
TIEP
SUBJECT SEES THE GAME AS AN OBSTACLE TO LEARNING

I mean, the whole idea is to make it fun and to learn something at the same time
PIEP

but what kids are gonna end up doing is they're gonna get frustrated, at least I am, they're going to click all over it, things aren't going to be making any sense to them, cause there's no explanation how to use it. Hmmm. Bad.
TIEE

PUZZLE PATTERNS GAME

Ok so now I'll do a puzzle.
TGD

Again, no explanation, so I click on something
TID

and I have to they match things.
COD

Now again we rotate. We rotate things around.
OD

I'm guessing that that's what they want you to do.
ON

Oh that's really weird. So over here you've got a yellow one. Now that won't match up with the blue one, but OK. Let's just assume that that's what they want you to do. (mumbling)Oh they want you to do a pattern.
CPD

Again, they don't explain. so. Oh, I have no idea.
TID

I have no idea what this game wants me to do so...Bye.
TGD
IF THE GAME IS TOO COMPLEX SUBJECT EXITS BECAUSE HIS FOCUS IS ON EVALUATION, NOT PLAY.

PUZZLE GAME

Alright so here I have to....OK. Oh I thought that each piece, had a specific, uhm, originally I thought that each piece had a specific function, and a specific picture assigned to each piece, but that's not the case.

TID

So far this is my favorite game along with the rockets. But I think that the biggest drawback in any of these games is that there's no guidance or explanation or nothing.
TIEE

I think that that's -- I just checked out that little saw blade, see what that is...
TGD

Sorry.
OF
Soon as it clears I'm going outta here.
OD

Well that's it for me.
OD

Unless these windows are something.
TGD

ART GAME

OK again, no explanation.
TID

Can you pick a color?
TGR

Oh, I don't know. I don't like that. Oh, I see, so you can fill it, so it fills with whatever color. So this is a tan grams or uh, that kind of thing, and I you can make a pattern, and oh, I see a little printer icon, which means I can print out my design if I want. Its OK, its Ok It doesn't do anything for me.
PIEN

I wouldn't buy it. I wouldn't buy it. I've seen lots of other software that works so much better, I wouldn't buy this.
TGEN

I think as a summary, I think that the graphics are very nice,
TAEN

I think that a lot of other programs give you more things to click on that are just fun to click on just for fun sake.
TGEE

I think a lot of these things, once you click on them, they do something that you completely don't expect, and don't know how to get back to the original screen, there's not enough explanation, and I think there's too much convoluted thought that goes into it.
TIEE

I'm a little worried, I think that its marketed to like third, or maybe fourth grade but uh,
PID

and I might be wrong, I just don't think, I don't know too many kids that would want to stick with the frustration level would be very high in having to kind of grope around and figure out what to do and possibly not being able to do it. So, I wouldn't buy it.
PAEE

OUTNUMBERED

So now we're looking for the professor.
TGD

OK this is similiar
ON

Right.
OF

Ok so now, I go in, OK so I gotta sign in.
TGD

Ok hold on a sec, now what I used to do
OF

so now you get to bump into things to find out if you're going to get the clue or not
TGD

(Reads the problem)

So its twenty eight.
CPD

Twenty-eight. enter
CPD

So I have to click enter?
TGR

Ok. so now I continue. I go to the next thing.
TGD

There must be a door here.
TGD

Now I know he's got to get energy somehow cause his energy is dwindling.
TGD

I like these buttons better.
TIEN

There's energy. Let's see if he can get energy.
TGD

No. OK

OF

So I don't know how I can get more energy.
TGD

Ok so I have to bump into everything again.
TGD

You what I never liked about these games, you have to keep bumping into everything.
TGEE

(Reads problem)
So you have to first of all make sure you look in the right column for Call's last month. So GO GO Shop is 66, and Green Peas is 51, so...15, so you can do it...I have to assume that this calculator is here so you can use it.
CPD

minus..uhmm...51 equal, 15 enter.
CPD

So now they give you a calculator to use. I'm not sure I'm happy with that. I'm not sure I'm happy with that at all because I wonder if that really helps kids invent new ways to figure out their own solutions, I think it just does it for them.
PIEE
SUBJECT CONSIDERS THE EDUCATIONALLY VALID ADVANTAGES AND DISADVANTAGES TO THE CALCULATOR

Yes, they have to know to do 66 minus 15 but then the actual subtraction is not for them to do, the calculator does it, so, you know, you have to decide what level of problem solving you want the kids to engage in, and whether that satisfies you that they're supplying a calculator.
PIEE

Cause I would almost, maybe if this thing gave you enough time I'd almost rather they have manipulatives there, so that they could figure it out using some blocks or whatever it is or even on their fingers if they have to, not to have some other machine do it for them. I don't think that teaches them anything.
PIEE
SUBJECT CONSIDERS PREFERABLE ALTERNATIVES TO HOW THE

INSTRUCTION IS PRESENTED.

Again, they can't go in here so,
TGD

so I've gotta go back out of the room, and again,
TGD

Come on, did it lock up?
TGR

Here we are bumping into stuff.
TGD
(reads problem)

So we got 138 pairs of baseball socks, 135 pairs have the colors of the SG team, so its 138 minus 135 so its obviously 3,
CPD

so obviously a kid has to figure out, to know that he has to look up the baseball sock team, and he's got to be able to know to subtract the 135 from that, and then he'll do it on the calculator again, so its 3 the answer, good job, continue.
COD

Uhm,
OF

oh there it is.
TGD

This doesn't do anything for me.
PAEN

No. I don't want to go there. Come on.
TGD
(Reads problem)

So 77 plus 75. Ok so its 140, 152, from here to here, from here to there, (unintelligible) which I learned.
CPD

(mumbling about the problem)

that's right, continue.
OF

Meanwhile, I don't seem to be getting any clues.
TGD

You've got to zap the TV and you can do that by hitting the spacebar.

what do you mean.
TGR

When he comes on again..

Oh that guy.
OF

Did they say that in the clues that you have to zap the
TIR

yeah, use the spacebar

Well, I didn't really realize. I'll see if I can get him again.
TGD

So now what.
OF

Oh I should have done that too.
TGD
(does the drill)

Now I get one clue. OK.
TGD

Doesn't do anything for me.
PAEN

I think that (unintelligible) you can waste the time again, alright alright.
PIEN

How do you turn the sound down?
TAR

You can set the preferences.

Oh no, I'm not gonna bother.
ON

OK anyway, problems I found with this is that yeah, its a really great for doodling and having a lot of time in which you're just fumbling around, again this is another game where you fumble fumble fumble fumble and you really don't know where to start looking,
TID

you know its nice that you, kids like to zap things, so I know why they put a zap thing in there but uh,
PAD

they don't explain how to get the energy,
TID

uhm and, I know a lot of kids just play the games to play them,
PAD

I know my son, when he was in school, was playing uh, what was that game? Oregon Trail, and he just played, you know, he didn't follow any of the strategies, that didn't explain to him what to do, and it wasn't until Oregon Trail 2 or Oregon Trail 3 that they tried to actually address those problems uh because a kid could just, you know, pick
anything off the stores list, have you played that game?
PID
SUBJECT HAS EXPERIENCE WITH OTHER EDUCATIONAL SOFTWARE AND CAN DRAW COMPARISONS.

yes

yeah a kid could take anything off the stores list just to get going and very often I saw that the kids in my class just took anything, they didn't bother to think what am I taking, why, what's the purpose to it, how far will it get me, so the strategy is what we are, the point of imbuing strategy in children, and getting them to think about what they're doing,
PIEP

is warped, because all they're doing is grabbing something, anything, so its completely an aimless exercise
PIEE

and this has a lot of this also, its completely aimless

PIEE

and you figure in a class, where they're in a computer lab, where they're given forty minutes to play, or they're, or in a class, where they're given one, they usually take a list and take turns but they don't have 20 computers in every classroom so in a classroom, predictably you usually have anywhere between two and four computers, so that usually that means that they have to ration the computers, usually they have a schedule for which day the kid gets the computer, like today's your computer day Johnny, so Johnny spends 45 minutes ambling through this one program, either getting frustrated and turning it down and going on to something else, or not getting frustrated and just sitting and playing.
PFEE
SUBJECT ADDRESSES THE PRACTICAL CONSTRAINTS OF USING THE SOFTWARE IN A CLASSROOM

You know I don't think there's enough math in there to,
COEE

I don't think there's enough uh, I don't want them to just be given problems,
PIEE

but I don't think there's enough of here's a clue, solve it to get to the to solve the puzzle or you know, Here's the prize, kind of thing,
PAEE

I think there's a lot of walking around and a lot of zapping, and I don't think that does, I don't think that does an awful lot for the kid for his learning prospects,
PIEE

I don't think the progress ratio is really high on it because he's not problem solving for the purpose of, he's not bumping into things for the purpose of,
PAEP

it takes too long to get a clue, too long to get anything,
PAEE

so in that 45 minute period how much is he getting out of it?
PAR

By the time he gets back to it, the following week, when its his time again, he's lost completely forgot where he is, you know, forget it.
PIEE

PROBES
Mathworkshop

1)
well, I can tell you what's intended by it, I think that they're intended to uh, let's see I'm trying to remember what the thing is, what do you call it, algorithms for the bowling, is what's intended,
COD

and I think that may do it because there's an immediate reward,
PAEN

But then again, there's no instruction unless you ask for it, there's no introduction on how to use it
TID

But there's immediate gratification cause either you get it or you didn't, if you miss the answer you get the right answer right away--instant gratification--or uh, if you get the answer correct you get a pin which is instant gratification, so the child gets constant gratification so I think that particular game works, uhm
PAEE

what do you call it, uhm, I think I said I liked the rocket part, its strategies,
PIEE

and I think it's take one two or three times for the kid to play it to realize that he's gotta figure how many to leave her with so that she can't shoot em all off. So he's left with something. So I think that the strategies thing might work and I think that its exciting
PIEE

...I don't know how many times a kid can see a rocket go off before he gets bored, but OK.
PAD

3)
Oh I definitely those are spatial.
COD

Definitely those are spatial
COD
so you can see if the kid can either match patterns, be able to rotate them in their mind, try to figure out whether they need rotating or not, and to be able to figure out whether they need to drag them onto the painting or not. I mean that's all tactile, that's all spatial, and again, its also matching, so uh, you know, its imaging, and I think that the whole objective for that is being able to satisfy their spatial needs.
COD
(met?)

Mildly, again I think that what retards their progress
PAD

is not having any uh, not having any initial instructions.
TID

5)
oh awful from the beginning,
ON

again no instruction,
TID

obviously it intended to reinforce the idea of parts of a whole and the different methods by which you can get to a whole, or how many, how many whole pieces whether its one whole or two whole or a fraction thereof, to be able to, to do multiple algorithms, to get to those concepts,
PIEP

but I don't think it does it because the game actually gets in the way of bearing that out.
PIEE

If an explanation, if a decent explanation was offered, here's how you play this game, and here's level one, level two, level three, level four, I think that that probably would have worked out better,
TIEE

and again, the game gets in the way, and I don't think that you learn anything,
PIEE

I think I would have just clicked exit and gone onto something else which I enjoyed.
PAEE

6)
Oh the rockets again I think were uhm strategies, in much the same way as an adult game of 21, the card game 21 would be, I think that the rockets do the same thing on a kids' level,
COEE

but then again, I think that was probably one of the better games, because it gives you an immediate visualization of what's left, what's there,
PIEE

I think what's bad about the game is that uhm is that you have to kind of guess that where it says number of rockets is the amount of rockets that are on the table to start with, its not how many rockets you can launch or you know, I think its very confusing the way that they laid it out,
TIEE

and I think that part of it gets in the way
ON

but on the other hand there's more of the ability to click things and have something happen, and then you can establish a pattern in your mind for what's supposed to occur in the game which is what it didn't do, which is what didn't happen with the fraction game.
TPEE

7)
I thought that was awful.
ON

Then again I think it was matching, it was matching it was also spatial, in might have been patterning in there
COD

but I couldn't get the pattern, because usually they'll

start off, what I like about some games is that they'll start off with an easy one, you get the idea you increase the complexity of it so you build you build on your initial idea of how does this work and then you can build on that more complex and more complex and more complex and it can go from the concrete to the abstract which is doesn't do. Its completely abstract to start out with and I don't think that's a good idea.
PIEP

8)
I know what that's for, that's basically for spatial and what do you call, visualization,
COD

I mean a lot of other games use it where they ask you to make a reversed tangram, or or or, they'll give you a pattern and they'll ask you to make a reverse pattern. In this case what I think they're asking you to do is to make your own pattern, and that notice that maybe if you put some of the triangles together they'll form a square or a rectangle or they'll form another geometric shape and then you'll start, and then when you realize that you'll start putting more of those shapes together and then uh, being able to pick whatever color,
COD

and being able to print out is a reward, you know, here's my picture, now I've made it and look what I've done,
PAD

so I think that the intent is is is geometry and spatial, and shapes and shape identitification.
COD

Uh, I think that it might work, I think its one of the better choices just because there's a color reward and you can print out anything you want,
PAEE

and then, you know, I think after you investigate it for a while, you'll start to make those patterns in your mind, so I think that's fine. OK.
PIEE

9)
Well I think she's nice and friendly, but she doesn't she doesn't give me any kind of direction and I get angry at

her.
TIEE

10)
there's, first of all, I hate the fact that uh, that some of the games give you the answer right away, I don't think that's feedback, I think that prevents you from thinking of, from using strategies to find other ways of thinking how can I get the answer. So that uh, I don't like that,
PAEP

and uhm, as far as those other little buttons are concerned, I found out, the buttons with the faces on them? everytime I clicked on it, it didn't, you know it didn't, what some programs do is that when you put your cursor uh on a button, a radio button, whatever other kind of button, animated button, a little box will come up to tell you what the button's for. This doesn't do that, so in order to find out what its for, you have to click it.
TPEE

Once you've clicked it, something happens that you may not expect and it may change the game, it may...you have no idea what's gonna go on, and then you have to first figure out how to get back where you were. Don't like it.
TIEE

11)
As far as this first program, Math Workshop? I don't know what you mean by technical.
ON

In other words how the program itself works apart from its instructional...

You mean the ergonaumics of it? I think that its confusing,
TGEN

I think that having the flashy things say go did not necessarily, I think that usually, you know we've discovered that children learn in a lot of different ways, and just having, if a kid is not a visual learner or if he doesn't learn if he's not a great reader, the flashing go is not gonna really help him a lot
PIEP

I think that combined...if Polly would have said, touch the flashing go button to do such and such that might have helped, you know, you know I think there are kids, especially if you have kids with learning disabilities, or autism, or something like that, they're not gonna make the connection, you know, they might see it as OG, and so its flashing; there are a lot of other things moving around, so what?
PIEP

12)
Well I think that basically they're trying to cover a lot of different concepts and I think that's a strength,
COEN

but I think completely weakened by the fact that uh, so much, so much of the stuff is uh counter productive to learning so I wouldn't buy it for that reason.
PIEE

13) in class
I wouldn't use this software
ON

OUTNUMBERED

1)
Oh basic algorithms, algorithm functions, that's it. Oh I shouldn't say that's it, let me go back. There's also some problem solving, but its basic problem solving,
COD

It depends on the level of the child,
PID

but getting to the problems is uh getting to those problems is, takes too much time away from actually solving,
PIEE

they don't make actually solving the problem part of the math,
PIEN

the math is, its almost like there are two worlds--you have to find these other clues and in order to do that you have to go in this hallway and bump into a bunch of

stuff, maybe you'll get a problem to solve and uh, all those things are disjointed from each other, I don't think there's one big connective tissue that runs through all of it.
TGEE

2)
I didn't even know it was timed so I couldn't--I also think that timing what I 've learned is that there are some kids who really get off on being timed...its a challenge to them, there are other kids who get frustrated by it and they just turn right off and say forget it, I don't want to do this, and then there are other kids for whom timing is counter productive only because they, they're learning style may be such that they need more time and its not to say that they won't get it; the timing just doesn't do very much for them.
PIEP

5)
Well, I didn't like the fact that if I didn't do, if I looked the wrong way for the wrong second that my character would get killed or knocked on his rear end. I think that's purely just the manufacturer's attempt to try to integrate a learning experience with a pinball machine.
PAEE

6) tech strengths
Uhm, I think that uh, I think that basically I don't see too many technical strengths
ON

I think its mostly a game for kids to be amused by,
TGEN

I think there's very little learning going on,
PAEN

I think it takes too long to get to each learning point,
PIEN

uh, because the guy bumping in and being chased by this thing has nothing to do with let's solve this problem, or here's a problem,
PIEE

in order to, in order for us to get the prize whatever, what do you call it, whatever that satisfying thing is, uh its not part of the bumping into the walls, its part of solving the problem that comes up every once in a while when you finally discover what you're supposed to bump into.

7)
I think that all they're offering is algorithms and some basic problem solving,
COD

and I don't think there's enough of it gotten to quickly enough to have much use.
PIEE

8) use
I wouldn't use it.
ON

APPENDIX G

PROBE/INTERVIEW QUESTIONS

PROBES
Math Workshop

1. What specific math learning objectives would be likely to be met by using Math Workshop? Include any concepts that you believe would be conveyed to students using the software.

2. What do you think are the intended learning objectives for the bowling game? Do you think they would be met? Explain.

3. What do you think are the intended learning objectives for the puzzle patterns game? Do you think they would be met? Explain.

4. What do you think are the intended learning objectives for the super stickers game? Do you think they would be met? Explain.

5. What do you think are the intended learning objectives for the rhythm shop game? Do you think they would be met? Explain.

6. What do you think are the intended learning objectives for the rockets game? Do you think they would be met? Explain.

7. What do you think are the intended learning objectives for the pattern windows game? Do you think they would be met? Explain.

8. What do you think are the intended learning objectives for the art game? Do you think they would be met? Explain.

9. What is your impression of the main character, Polly? Explain.

10. Comment on the program's use of feedback.

11. Comment on the program's technical strengths and weaknesses.

12. Discuss any instructional strengths/weaknesses in

the program.

13. What suggestions can you make for how you might use this software in your classroom? Would you use it in conjunction with any other materials?

Outnumbered

1. What mathematical learning objectives would be likely to be met by using OutNumbered? Include any concepts that you believe would be conveyed to students using the software.

2. Is there any instructional significance to the fact that the game is timed? Explain.

3. What is your impression of the main character Telly?

4. What are students learning from the decoder game in the program?

5. Comment on the program's use of feedback.

6. Comment on the program's technical strengths and weaknesses.

7. Discuss any instructional strengths/weaknesses in the program.

8. What suggestions can you make for how you might use this software in your classroom? Would you use it in conjunction with any other materials?

APPENDIX H

TABLES

Table H1

Frequencies and Percents of Statements Generated By Novices and Teachers Across Major Focus Categories (Software Combined)

GROUP	Count / Row Pct	T	C	P	OD	OF	ON	Row Total
Novice		493 37.8	240 18.4	130 10.0	99 7.6	242 18.6	100 7.7	1304 46.2
Teacher		816 53.7	169 11.1	238 15.7	61 4.0	171 11.3	64 4.2	1519 53.8
	Column Total	1309 46.4	409 14.5	368 13.0	160 5.7	413 14.6	164 5.8	2823 100.0

Chi-Square	Value	DF	Significance
Pearson	137.27722	5	.00000
Likelihood Ratio	138.04073	5	.00000

Minimum Expected Frequency - 73.907

Table H2

Frequencies and Percents of Statements Generated By Novices and Teachers Across Major Focus Categories ("Other" Category Excluded)

Software: Math Workshop

GROUP (Count / Row Pct)	T	C	P	Row Total
Novice	282 54.9	147 28.6	85 16.5	514 41.4
Teacher	470 64.5	94 12.9	165 22.6	729 58.6
Column Total	752 60.5	241 19.4	250 20.1	1243 100.0

Chi-Square	Value	DF	Significance
Pearson	48.51894	2	.00000
Likelihood Ratio	47.93358	2	.00000

Minimum Expected Frequency - 99.657

Table H2 (Continued)

Frequencies and Percents of Statements Generated By Novices and Teachers Across Major Focus Categories (Other Category Excluded)

Software: Outnumbered

GROUP (Count / Row Pct)	T	C	P	Row Total
Novice	211 60.5	93 26.6	45 12.9	349 41.4
Teacher	346 70.0	75 15.2	73 14.8	494 58.6
Column Total	557 66.1	168 19.9	118 14.0	843 100.0

Chi-Square	Value	DF	Significance
Pearson	16.85041	2	.00022
Likelihood Ratio	16.62304	2	.00025

Minimum Expected Frequency - 48.852

Table H3

Frequencies and Percents of Explanation Statements Generated by Novices and Teachers (Software Combined)

GROUP	Count / Row Pct	OTHER	D	EN	EE	EP	R	Row Total
Novice		441 33.8	445 34.1	112 8.6	94 7.2	2 .2	210 16.1	1304 46.2
Teacher		296 19.5	712 46.9	160 10.5	163 10.7	14 .9	174 11.5	1519 53.8
	Column Total	737 26.1	1157 41.0	272 9.6	257 9.1	16 .6	384 13.6	2823 100.0

Chi-Square	Value	DF	Significance
Pearson	113.79974	5	.00000
Likelihood Ratio	115.27240	5	.00000

Minimum Expected Frequency - 7.391

Table H4

Frequencies and Percents of Statements Generated By Novices and Teachers in the Technical Category: Variable Clusters and Explanation Categories Expanded

Software: Math Workshop

GROUP (Count / Row Pct)	TAD	TAEN	TAEE	TID	TIEN	TIEE	TIR	TKD	TKEN	TKEE	TKR	Row Total
Novice	14 5.0	9 3.2	7 2.5	1 .4	3 1.1	10 3.5	3 1.1	1 .4			2 .7	282 37.5
Teacher	10 2.1	11 2.3	4 .9	10 2.1	17 3.6	13 2.8	2 .4	10 2.1	5 1.1	3 .6	2 .4	470 62.5
Column Total	24 3.2	20 2.7	11 1.5	11 1.5	20 2.7	23 3.1	5 .7	11 1.5	5 .7	3 .4	4 .5	752 100.0

(Continued)

GROUP (Count / Row Pct)	TGD	TGEN	TGEE	TGR	Row Total
Novice	114 40.4	25 8.9	7 2.5	86 30.5	282 37.5
Teacher	260 55.3	27 5.7	5 1.1	91 19.4	470 62.5
Column Total	374 49.7	52 6.9	12 1.6	177 23.5	752 100.0

Chi-Square	Value	DF	Significance
Pearson	48.37288	14	.00001
Likelihood Ratio	52.88420	14	.00000

Minimum Expected Frequency - 1.125
Cells with Expected Frequency < 5 - 12 OF 30 (40.0%)

Table H4 (continued)

Frequencies and Percents of Statements Generated By Novices and Teachers in the Technical Category: Variable Clusters and Explanation Categories Expanded

Software: Outnumbered

GROUP (Row Pct)	TAD	TAEN	TAEE	TID	TIEN	TIEE	TIR	TKD	TKEN	TKEE	TKR	Row Total
Novice	5 2.4	4 1.9		3 1.4		1 .5	1 .5	3 1.4	1 .5	1 .5	1 .5	211 37.9
Teacher	3 .9	5 1.4	3 .9	13 3.8	10 2.9	12 3.5	5 1.4	8 2.3	1 .3	3 .9	4 1.2	346 62.1
(Continued) Column Total	8 1.4	9 1.6	3 .5	16 2.9	10 1.8	13 2.3	6 1.1	11 2.0	2 .4	4 .7	5 .9	557 100.0

Row Pct	TGD 1041	TGEN 1042	TGEE 1043	TGR 1045	Row Total
Novice	101 47.9	11 5.2		79 37.4	211 37.9
Teacher	219 63.3	11 3.2	8 2.3	41 11.8	346 62.1
Column Total	320 57.5	22 3.9	8 1.4	120 21.5	557 100.0

Chi-Square	Value	DF	Significance
Pearson	71.96134	14	.00000
Likelihood Ratio	79.43270	14	.00000

Minimum Expected Frequency - .758
Cells with Expected Frequency < 5 - 18 OF 30 (60.0%)

Table H4 (continued)

Frequencies and Percents of Statements Generated By Novices and Teachers in the Content Category: Variable Clusters and Explanation Categories Expanded

Software: Math Workshop

GROUP	Count / Row Pct	CID	CIEN	CSEP	CDD	CDEN	CDEE	CDR	CPD	CPR	COD	COEN	Row Total
Novice	Count				5	12		1	80	4	26	9	147
	Row Pct				3.4	8.2		.7	54.4	2.7	17.7	6.1	61.0
Teacher	Count	1	1	1	19	4	1		15		35	10	94
	Row Pct	1.1	1.1	1.1	20.2	4.3	1.1		16.0		37.2	10.6	39.0
(Continued)	Column Total	1	1	1	24	16	1	1	95	4	61	19	241
		.4	.4	.4	10.0	6.6	.4	.4	39.4	1.7	25.3	7.9	100.0

GROUP	Count / Row Pct	COEE	COR	Row Total
Novice	Count	5	5	147
	Row Pct	3.4	3.4	61.0
Teacher	Count	5	2	94
	Row Pct	5.3	2.1	39.0
	Column Total	10	7	241
		4.1	2.9	100.0

Chi-Square	Value	DF	Significance
Pearson	59.53004	12	.00000
Likelihood Ratio	65.15955	12	.00000

Minimum Expected Frequency - .390
Cells with Expected Frequency < 5 - 15 OF 26 (57.7%)

Table H4 (continued)

Frequencies and Percents of Statements Generated By Novices and Teachers in the Content Category: Variable Clusters and Explanation Categories Expanded

Software: Outnumbered

GROUP	Count / Row Pct	CAR	CDD	CDEN	CPD	CPEE	CPR	COD	COEN	COEE	COR	Row Total
Novice	Count		4	2	58	1	7	16	1	2	2	93
	Row Pct		4.3	2.2	62.4	1.1	7.5	17.2	1.1	2.2	2.2	55.4
Teacher	Count	1	5	1	50		2	14	1	1		75
	Row Pct	1.3	6.7	1.3	66.7		2.7	18.7	1.3	1.3		44.6
Column Total		1	9	3	108	1	9	30	2	3	2	168
		.6	5.4	1.8	64.3	.6	5.4	17.9	1.2	1.8	1.2	100.0

Chi-Square	Value	DF	Significance
Pearson	6.42669	9	.69657
Likelihood Ratio	8.07234	9	.52687

Minimum Expected Frequency - .446
Cells with Expected Frequency < 5 - 16 OF 20 (80.0%)

Table H5

Frequencies and Percents of Statements Generated By Novices and Teachers in the Pedagogical Category: Variable Clusters and Explanation Categories Expanded

Software: Math Workshop

GROUP	Count / Row Pct	PAD	PAEE	PAEE	PAEP	PAR	PFD	PFEN	PFEE	PFR	PID	PIEN	Row Total
Novice	Count	5	8	20		2			1		3	14	85
	Row Pct	5.9	9.4	23.5		2.4			1.2		3.5	16.5	34.0
Teacher	Count	14	10	26	2	11	4	2	1	1	11	28	165
	Row Pct	8.5	6.1	15.8	1.2	6.7	2.4	1.2	.6	.6	6.7	17.0	66.0
(Continued)	Column Total	19	18	46	2	13	4	2	2	1	14	42	250
		7.6	7.2	18.4	.8	5.2	1.6	.8	.8	.4	5.6	16.8	100.0

GROUP	Count / Row Pct	PIEE	PIEP	PIR	Row Total
Novice	Count	20	2	10	85
	Row Pct	23.5	2.4	11.8	34.0
Teacher	Count	46	5	4	165
	Row Pct	27.9	3.0	2.4	66.0
	Column Total	66	7	14	250
		26.4	2.8	5.6	100.0

Chi-Square	Value	DF	Significance
Pearson	20.31687	13	.08759
Likelihood Ratio	22.85387	13	.04346

Minimum Expected Frequency - .340
Cells with Expected Frequency < 5 - 15 OF 28 (53.6%)

Table 5H (continued)

Frequencies and Percents of Statements In the Pedagogical Category: Cluster Variables and Explanation Attributes Expanded

Outnumbered

GROUP	Count / Row Pct	PAD	PAEE	PAEE	PAEP	PAR	PFD	PFR	PID	PIEN	PIEE	PIEP	Row Total
Novice	Count	4	4	12		1		1	2	9	7		45
	Row Pct	8.9	8.9	26.7		2.2		2.2	4.4	20.0	15.6		38.1
Teacher	Count	3	6	13	1	3	1		7	10	19	5	73
	Row Pct	4.1	8.2	17.8	1.4	4.1	1.4		9.6	13.7	26.0	6.8	61.9
(Continued)	Column Total	7	10	25	1	4	1	1	9	19	26	5	118
		5.9	8.5	21.2	.8	3.4	.8	.8	7.6	16.1	22.0	4.2	100.0

GROUP	Count / Row Pct	PIR	Row Total
Novice	Count	5	45
	Row Pct	11.1	38.1
Teacher	Count	5	73
	Row Pct	6.8	
	Column Total	10	118
		8.5	100.0

Chi-Square	Value	DF	Significance
Pearson	11.98233	11	.36497
Likelihood Ratio	14.76370	11	.19357

Minimum Expected Frequency - .381
Cells with Expected Frequency < 5 - 15 OF 24 (62.5%)

Table H6

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

SUBID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
TAD	2 7.7	3 11.5		2 7.7	1 3.8	2 7.7		3 11.5					1 3.8			26 1.4
TAEN				3 14.3		1 4.8		2 9.5					1 4.8	2 9.5	1 4.8	21 1.2
TAEE		3 25.0			2 16.7	1 8.3		1 8.3							2 16.7	12 .7
TID							1 4.5									22 1.2
TIEN								1 4.0		1 4.0			1 4.0		2 8.0	25 1.4
TIEE		5 17.2		1 3.4		2 6.9				2 6.9					3 10.3	29 1.6
TIEP																2 .1
TIR								3 42.9								7 .4
TKD														1 9.1	1 9.1	11 .6
TKEN															2 40.0	5 .3
TKEE															1 33.3	3 .2
TKR					2 50.0											4 .2
TGD	4 1.0	1 .3	4 1.0	1 .3	8 2.0	11 2.8	11 2.8	4 1.0	21 5.3	8 2.0	3 .8	2 .5	11 2.8	25 6.3	11 2.8	398 21.8

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

SUBID / FULLCODE (Row Pct)	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
TAD	1 3.8					1 3.8			6 23.1	2 7.7				2 7.7		26 1.4
TAEN					1 4.8	1 4.8			4 19.0			4 19.0		1 4.8		21 1.2
TAEE									1 8.3			1 8.3		1 8.3		12 .7
TID	2 9.1			1 4.5	1 4.5			4 18.2	1 4.5	1 4.5				11 50.0		22 1.2
TIEN	5 20.0					3 12.0		2 8.0	5 20.0					4 16.0	1 4.0	25 1.4
TIEE	2 6.9					2 6.9		3 10.3	2 6.9			1 3.4		6 20.7		29 1.6
TIEP														2 100.0		2 .1
TIR	2 28.6														2 28.6	7 .4
TKD	4 36.4	1 9.1	1 9.1						2 18.2	1 9.1						11 .6
TKEN												3 60.0				5 .3
TKEE	1 33.3	1 33.3														3 .2
TKR												2 50.0				4 .2
TGD	35 8.8	4 1.0	17 4.3	17 4.3	6 1.5	16 4.0	17 4.3	5 1.3	75 18.8	13 3.3	5 1.3	31 7.8	8 2.0	15 3.8	9 2.3	398 21.8

Table H6 (continued)

<u>Frequencies and Percents of Statements Generated By Individual Subjects</u>

Software: Math Workshop

FULLCODE (Count / Row Pct) \ SUBID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
TGEN	2 3.6	1 1.8	4 7.1		1 1.8	5 8.9		2 3.6	3 5.4	3 5.4		1 1.8		3 5.4	3 5.4	56 3.1
TGEE	2 15.4			1 7.7		1 7.7			2 15.4					1 7.7		13 .7
TGR	8 4.3		6 3.2	1 .5	8 4.3	5 2.7	5 2.7	2 1.1	5 2.7	7 3.8	3 1.6	10 5.4	4 2.2	22 11.8	2 1.1	186 10.2
CID															1 100.0	1 .1
CIEN																1 .1
CSD																2 .1
CSEP																1 .1
CDD		1 4.2				1 4.2	1 4.2				1 4.2			1 4.2		24 1.3
CDEN	2 11.1		2 11.1		4 22.2						2 11.1			2 11.1		18 1.0
CDEE																2 .1
CDR	1 50.0															2 .1
CPD			2 1.9		33 30.8	13 12.1	7 6.5	5 4.7			6 5.6	1 .9	2 1.9	11 10.3		107 5.9
CPR			1 16.7			1 16.7						1 16.7		1 16.7		6 .3

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

SUBID Row Pct	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
TGEN	5		3	1	2		2	1	7			3		1	3	56
	8.9		5.4	1.8	3.6		3.6	1.8	12.5			5.4		1.8	5.4	3.1
TGEE		2		1			1		1					1		13
		15.4		7.7			7.7		7.7					7.7		.7
TGR	19	14	8	2	1	2	10	6	3	6	4	11	3	2	7	186
	10.2	7.5	4.3	1.1	.5	1.1	5.4	3.2	1.6	3.2	2.2	5.9	1.6	1.1	3.8	10.2
CID																1
																.1
CIEN		1														1
		100.0														.1
CSD															2	2
															100.0	.1
CSEP		1														1
		100.0														.1
CDD	1		2	1			1		4	1	4	4	1			24
	4.2		8.3	4.2			4.2		16.7	4.2	16.7	16.7	4.2			1.3
CDEN		1										1	2		2	18
		5.6										5.6	11.1		11.1	1.0
CDEE													1		1	2
													50.0		50.0	.1
CDR															1	2
															50.0	.1
CPD	1		2				2	1	1		2	3	3	11	1	107
	.9		1.9				1.9	.9	.9		1.9	2.8	2.8	10.3	.9	5.9
CPR														1	1	6
														16.7	16.7	.3

Table 6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

SUBID

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
COD	1 1.5	2 3.0	2 3.0			4 6.0	10 14.9	1 1.5		1 1.5		2 3.0		3 4.5	1 1.5	67 3.7
COEN						2 10.5	2 10.5	3 15.8			1 5.3	1 5.3				19 1.0
COEE						1 10.0		3 30.0				1 10.0				10 .5
COR						1 14.3	4 57.1									7 .4
PAD				1 4.5			2 9.1	1 4.5						1 4.5	1 4.5	22 1.2
PAEE		1 4.8				1 4.8	2 9.5	3 14.3	1 4.8						1 4.8	21 1.2
PAEE		6 11.8				6 11.8		2 3.9	2 3.9	2 3.9		2 3.9			2 3.9	51 2.8
PAEP																2 .1
PAR				1 7.7			1 7.7									13 .7
PFD																4 .2
PFEN																2 .1
PFEE										1 50.0						2 .1
PFR																1 .1

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

Row Pct \ SUBID	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
COD	4 6.0	2 3.0	5 7.5	1 1.5	4 6.0	1 1.5		1 1.5	4 6.0	2 3.0	2 3.0	7 10.4	1 1.5	3 4.5	3 4.5	67 3.7
COEN	1 5.3			2 10.5	2 10.5				2 10.5			3 15.8				19 1.0
COEE	1 10.0					2 20.0	1 10.0					1 10.0				10 .5
COR					1 14.3		1 14.3									7 .4
PAD				2 9.1	3 13.6			1 4.5	1 4.5	2 9.1	2 9.1		2 9.1		3 13.6	22 1.2
PAEE			3 14.3		1 4.8	1 4.8			3 14.3			1 4.8		3 14.3		21 1.2
PAEE	3 5.9	1 2.0	1 2.0	5 9.8	2 3.9	2 3.9	1 2.0			8 15.7		1 2.0		1 2.0	4 7.8	51 2.8
PAEP	1 50.0	1 50.0														2 .1
PAR				2 15.4		1 7.7	5 38.5			1 7.7	1 7.7		1 7.7			13 .7
PFD		1 25.0			2 50.0							1 25.0				4 .2
PFEN		1 50.0										1 50.0				2 .1
PFEE								1 50.0								2 .1
PFR		1 100.0														1 .1

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

FULLCODE (SUBID: Count / Row Pct)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
PID Count				1			1	1							2	21
PID Row Pct				4.8			4.8	4.8							9.5	1.2
PIEN Count	1	1			1	2	1	4	1	1		2			1	44
PIEN Row Pct	2.3	2.3			2.3	4.5	2.3	9.1	2.3	2.3		4.5			2.3	2.4
PIEE Count	1	3		2		2	2	2		6		2			1	72
PIEE Row Pct	1.4	4.2		2.8		2.8	2.8	2.8		8.3		2.8			1.4	3.9
PIEP Count										2						10
PIEP Row Pct										20.0						.5
PIR Count				2		2	2			1		2		1		14
PIR Row Pct				14.3		14.3	14.3			7.1		14.3		7.1		.8
OD Count	3	1	2		5	7	5	6		2	2	1	6	12		95
OD Row Pct	3.2	1.1	2.1		5.3	7.4	5.3	6.3		2.1	2.1	1.1	6.3	12.6		5.2
OF Count	6		5		15	17	5	8	9	14	9	23	4	24	1	247
OF Row Pct	2.4		2.0		6.1	6.9	2.0	3.2	3.6	5.7	3.6	9.3	1.6	9.7	.4	13.5
ON Count	4	3		1	7	8	5	9		6	1	2	3	10	4	115
ON Row Pct	3.5	2.6		.9	6.1	7.0	4.3	7.8		5.2	.9	1.7	2.6	8.7	3.5	6.3

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Math Workshop

SUBID Count Row Pct FULLCODE	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
PID	3 14.3				2 9.5			1 4.8			1 4.8	1 4.8	1 4.8	4 19.0	3 14.3	21 1.2
PIEN	7 15.9		3 6.8	1 2.3		2 4.5	1 2.3		7 15.9		3 6.8	2 4.5	1 2.3	1 2.3	1 2.3	44 2.4
PIEE	10 13.9		7 9.7	3 4.2	1 1.4	1 1.4	4 5.6	1 1.4	4 5.6	1 1.4	4 5.6	4 5.6	5 6.9	1 1.4	5 6.9	72 3.9
PIEP	2 20.0	1 10.0	1 10.0		1 10.0									2 20.0	1 10.0	10 .5
PIR	1 7.1	1 7.1					1 7.1					1 7.1				14 .8
OD	2 2.1	7 7.4	5 5.3		1 1.1	3 3.2	4 4.2	4 4.2	2 2.1			5 5.3	2 2.1	5 5.3	3 3.2	95 5.2
OF	15 6.1	18 7.3	1 .4	3 1.2	1 .4	2 .8	3 1.2	18 7.3	10 4.0	1 .4	1 .4	17 6.9	4 1.6	10 4.0	3 1.2	247 13.5
ON	6 5.2	3 2.6	8 7.0		1 .9	1 .9		2 1.7	6 5.2	4 3.5	1 .9	10 8.7	1 .9	6 5.2	3 2.6	115 6.3

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

FULLCODE (Count / Row Pct)	SUBID 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
TAD		3 37.5				1 12.5								1 12.5		8 .6
TAEN			1 11.1	2 22.2		1 11.1									1 11.1	9 .7
TAEE																3 .2
TAR																1 .1
TID		1 5.6		1 5.6				1 5.6								18 1.5
TIEN																11 .9
TIEE				1 7.7											2 15.4	13 1.1
TIR								1 11.1								9 .7
TKD				1 9.1										2 18.2		11 .9
TKEN									1 50.0							2 .2
TKEE									1 25.0						2 50.0	4 .3
TKR	1 20.0														1 20.0	5 .4
TGD	10 2.9	6 1.8	12 3.5	2 .6	15 4.4	11 3.2	3 .9	1 .3	8 2.3	6 1.8	8 2.3	1 .3	2 .6	16 4.7	7 2.0	342 27.6

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

SUBID

Row Pct

FULLCODE	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
TAD		2 25.0					1 12.5									8 .6
TAEN		1 11.1			1 11.1				2 22.2							9 .7
TAEE		2 66.7		1 33.3												3 .2
TAR														1 100.0		1 .1
TID	3 16.7	2 11.1		2 11.1	2 11.1	2 11.1	1 5.6	1 5.6						2 11.1		18 1.5
TIEN			1 9.1	2 18.2	2 18.2	3 27.3		1 9.1				1 9.1		1 9.1		11 .9
TIEE		1 7.7		3 23.1	1 7.7		1 7.7	1 7.7		3 23.1						13 1.1
TIR	1 11.1			2 22.2		1 11.1						1 11.1		1 11.1	2 22.2	9 .7
TKD	2 18.2	5 45.5								1 9.1						11 .9
TKEN												1 50.0				2 .2
TKEE		1 25.0														4 .3
TKR		2 40.0								1 20.0						5 .4
TGD	10 2.9	16 4.7	32 9.4	15 4.4	3 .9	7 2.0	18 5.3	9 2.6	48 14.0	7 2.0	11 3.2	29 8.5	7 2.0	18 5.3	4 1.2	342 [illegible].6

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

SUBID Row Pct (Count)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
TGEN						2 9.1				3 13.6	2 9.1		1 4.5	3 13.6	1 4.5	22 1.8
TGEE																9 .7
TGR	9 7.0		9 7.0	4 3.1	4 3.1	4 3.1	1 .8	1 .8	13 10.1	2 1.6	7 5.4	3 2.3	1 .8	21 16.3	2 1.6	129 10.4
CAR																1 .1
CDD											1 11.1			3 33.3		9 .7
CDEN						1 33.3								1 33.3		3 .2
CPD	2 1.8		2 1.8		12 10.5	6 5.3	1 .9	7 6.1	6 5.3			4 3.5	1 .9	17 14.9	4 3.5	114 9.2
CPEE								1 100.0								1 .1
CPR			1 11.1			1 11.1			1 11.1					4 44.4	1 11.1	9 .7
COD	1 3.2	4 12.9	1 3.2			2 6.5	3 9.7		2 6.5					3 9.7		31 2.5
COEN											1 33.3					3 .2
COEE	2 50.0															4 .3
COR														2 100.0		2 .2

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

SUBID

Row Pct	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
TGEN		2 9.1	3 13.6	1 4.5				2 9.1	1 4.5	1 4.5						22 1.8
TGEE		3 33.3		1 11.1		1 11.1		1 11.1		1 11.1		1 11.1		1 11.1		9 .7
TGR	1 .8	14 10.9	4 3.1	2 1.6		1 .8	6 4.7	4 3.1	1 .8	2 1.6	1 .8	3 2.3		3 2.3	6 4.7	129 10.4
CAR		1 100.0														1 .1
CDD					1 11.1	1 11.1			1 11.1	1 11.1			1 11.1			9 .7
CDEN					1 33.3											3 .2
CPD	2 1.8	7 6.1	10 8.8	5 4.4	2 1.8		2 1.8		1 .9		7 6.1	5 4.4	5 4.4	6 5.3		114 9.2
CPEE																1 .1
CPR		1 11.1														9 .7
COD	3 9.7		3 9.7		1 3.2		1 3.2	4 12.9	1 3.2			1 3.2		1 3.2		31 2.5
COEN			1 33.3												1 33.3	3 .2
COEE	1 25.0													1 25.0		4 .3
COR																2 .2

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

SUBID

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total	
PAD			1 11.1					1 11.1	1 11.1						1 11.1	1 11.1	9 .7
PAEE						1 8.3			1 8.3	1 8.3		1 8.3					12 1.0
PAEE		1 3.7	4 14.8	1 3.7	1 3.7		3 11.1		1 3.7	1 3.7						3 11.1	27 2.2
PAEP																	2 .2
PAR					1 20.0												5 .4
PFD																	1 .1
PFEE																	1 .1
PFR								1 100.0									1 .1
PID				1 10.0					1 10.0								10 .8
PIEN		1 5.0		1 5.0	1 5.0	1 5.0				4 20.0					1 5.0	2 10.0	20 1.6
PIEE		1 2.9		1 2.9	2 5.7		1 2.9				1 2.9				1 2.9	7 20.0	35 2.8
PIEP																	6 .5
PIR		1 10.0		1 10.0	2 20.0										1 10.0		10 .8

Table H6 (continued)

Frequencies and Percents of Statements Generated By Individual Subjects

Software: Outnumbered

	SUBID 16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Total
PAD								1 11.1	1 11.1					2 22.2		9 .7
PAEE			4 33.3						2 16.7					2 16.7		12 1.0
PAEE		2 7.4		2 7.4		1 3.7		3 11.1		2 7.4				2 7.4		27 2.2
PAEP						1 50.0								1 50.0		2 .2
PAR						1 20.0	1 20.0		1 20.0					1 20.0		5 .4
PFD								1 100.0								1 .1
PFEE														1 100.0		1 .1
PFR																1 .1
PID	1 10.0			1 10.0	1 10.0			4 40.0						1 10.0		10 .8
PIEN	2 10.0	1 5.0	1 5.0			1 5.0				2 10.0		1 5.0		1 5.0		20 1.6
PIEE		1 2.9	1 2.9	4 11.4			2 5.7	1 2.9	1 2.9	2 5.7				8 22.9	1 2.9	35 2.8
PIEP			2 33.3							3 50.0				1 16.7		6 .5
PIR	1 10.0	2 20.0					2 20.0									10 .8

Table H6 (continued)

<u>Frequencies and Percents of Statements Generated By Individual Subjects</u>

Software: Outnumbered

FULLCODE (Count / Row Pct)	SUBID 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Row Total
OD	3	1	5		2	5	2	6	1	1	6	2		13	4	75
	4.0	1.3	6.7		2.7	6.7	2.7	8.0	1.3	1.3	8.0	2.7		17.3	5.3	6.1
OF	13	1	7	2	14	10	3	6	17	3	1	5	1	20	4	188
	6.9	.5	3.7	1.1	7.4	5.3	1.6	3.2	9.0	1.6	.5	2.7	.5	10.6	2.1	15.2
ON	3		2	3	1	1	1	4	3		2	4		17		62
	4.8		3.2	4.8	1.6	1.6	1.6	6.5	4.8		3.2	6.5		27.4		5.0

FULLCODE (Count / Row Pct)	SUBID 16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Row Total
OD		5	5	2	1	1	4			1	1	2			2	75
		6.7	6.7	2.7	1.3	1.3	5.3			1.3	1.3	2.7			2.7	6.1
OF	2	14	20	2		3	1	1	2	3	6	9	9	7	2	188
	1.1	7.4	10.6	1.1		1.6	.5	.5	1.1	1.6	3.2	4.8	4.8	3.7	1.1	15.2
ON	2	6	3				3	1	1			1		2	2	62
	3.2	9.7	4.8				4.8	1.6	1.6			1.6		3.2	3.2	5.0
Column Total	31	91	90	45	16	24	43	35	63	30	26	55	22	64	20	1237
	2.5	7.4	7.3	3.6	1.3	1.9	3.5	2.8	5.1	2.4	2.1	4.4	1.8	5.2	1.6	100.0

Table H7

Frequencies and Percents of Statements From Educational Technologists (Across Major Focus Categories)

Software: Math Workshop

GROUP	Count / Row Pct	T	C	P	OD	OF	ON	Row Total
ET	Count	68	26	29	8	13	9	153
	Row Pct	44.4	17.0	19.0	5.2	8.5	5.9	100.0
	Column Total	68	26	29	8	13	9	153
		44.4	17.0	19.0	5.2	8.5	5.9	100.0

Software: Outnumbered

GROUP	Count / Row Pct	T	C	P	OD	OF	ON	Row Total
ET	Count	39	9	21	2	9	4	84
	Row Pct	46.4	10.7	25.0	2.4	10.7	4.8	100.0
	Column Total	39	9	21	2	9	4	84
		46.4	10.7	25.0	2.4	10.7	4.8	100.0

Table H8

Frequencies and Percents of Statements Generated by Educational Technologists (Variable Clusters and Explanations Expanded)

Software: Math Workshop

GROUP	Count / Row Pct	TAD	TAEN	TAEE	TID	TIEN	TIEE	TIEP	TIR	TGD	TGEN	TGEE	Row Total
ET	Count	2	1	1	11	5	6	2	2	24	4	1	153
	Row Pct	1.3	.7	.7	7.2	3.3	3.9	1.3	1.3	15.7	2.6	.7	100.0
(Continued)	Column Total	2	1	1	11	5	6	2	2	24	4	1	153
		1.3	.7	.7	7.2	3.3	3.9	1.3	1.3	15.7	2.6	.7	100.0

GROUP	Count / Row Pct	TGR	CSD	CDEN	CDEE	CDR	CPD	CPR	COD	PAD	PAEE	PAEE	Row Total
ET	Count	9	2	2	1	1	12	2	6	3	3	5	153
	Row Pct	5.9	1.3	1.3	.7	.7	7.8	1.3	3.9	2.0	2.0	3.3	100.0
(Continued)	Column Total	9	2	2	1	1	12	2	6	3	3	5	153
		5.9	1.3	1.3	.7	.7	7.8	1.3	3.9	2.0	2.0	3.3	100.0

GROUP	Count / Row Pct	PID	PIEN	PIEE	PIEP	OD	OF	ON	Row Total
ET	Count	7	2	6	3	8	13	9	153
	Row Pct	4.6	1.3	3.9	2.0	5.2	8.5	5.9	100.0
	Column Total	7	2	6	3	8	13	9	153
		4.6	1.3	3.9	2.0	5.2	8.5	5.9	100.0

Table H8 (continued)

Frequencies and Percents of Statements Generated by Educational Technologists

(Variable Clusters and Explanations Expanded)

Software: Outnumbered

GROUP	Count / Row Pct	TAR	TID	TIEN	TIR	TGD	TGEE	TGR	CPD	COD	COEN	COEE	Row Total
ET	Count	1	2	1	3	22	1	9	6	1	1	1	84
	Row Pct	1.2	2.4	1.2	3.6	26.2	1.2	10.7	7.1	1.2	1.2	1.2	100.0
(Continued)	Column Total	1	2	1	3	22	1	9	6	1	1	1	84
		1.2	2.4	1.2	3.6	26.2	1.2	10.7	7.1	1.2	1.2	1.2	100.0

GROUP	Count / Row Pct	PAD	PAEE	PAEE	PAEP	PAR	PFEE	PID	PIEN	PIEE	PIEP	OD	Row Total
ET	2	2	2	2	1	1	1	1	1	9	1	2	84
		2.4	2.4	2.4	1.2	1.2	1.2	1.2	1.2	10.7	1.2	2.4	100.0
(Continued)	Column Total	2	2	2	1	1	1	1	1	9	1	2	84
		2.4	2.4	2.4	1.2	1.2	1.2	1.2	1.2	10.7	1.2	2.4	100.0

GROUP	Count / Row Pct	OF	ON	Row Total
ET	2	9	4	84
		10.7	4.8	100.0
	Column Total	9	4	84
		10.7	4.8	100.0

Table H9

Frequencies and Percents of Explanation Statements Generated by Educational Technologists

Software: Math Workshop

GROUP	Count / Row Pct	Other	D 1	EN 2	EE 3	EP 4	R 5	Row Total
ET	Count	30	67	17	20	5	14	153
	Row Pct	19.6	43.8	11.1	13.1	3.3	9.2	100.0
Column Total		30	67	17	20	5	14	153
		19.6	43.8	11.1	13.1	3.3	9.2	100.0

Software: Outnumbered

GROUP	Count / Row Pct	Other	D 1	EN 2	EE 3	EP 4	R 5	Row Total
ET	Count	15	34	5	14	2	14	84
	Row Pct	17.9	40.5	6.0	16.7	2.4	16.7	100.0
Column Total		15	34	5	14	2	14	84
		17.9	40.5	6.0	16.7	2.4	16.7	100.0

References

Alexander, P. A. (1992). Domain knowledge: Evolving themes and emerging concerns. Educational Psychologist, 27, 33-51.

Bean, R. M., Fulmer, D., Zigmond, N., Grumet, J. V. (1995). How experienced teachers think about their teaching: Their focus, beliefs, and types of reflection. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, CA, April 1995.

Berliner, D. C. (1991). Educational psychology and pedagogical expertise: New finding and new opportunities for thinking about training. Educational Psychologist, 26, 145-155.

Birnbaum, M. H. (1977). Expectancy and judgment. In N. J. Castellan, Jr., D. B. Pisoni, & G. R. Potts (Eds.), Cognitive Theory (Vol 2) (pp. 107-117). Hillsdale, NJ: Erlbaum.

Bitter, G. (1993). Using a microcomputer in the classroom. Boston, MA: Allyn and Bacon.

Borko, H., & Livingston, C. (1989). Cognition and improvisation: Differences in mathematics instruction by expert and novice teachers. American Educational Research Journal, 26, 473-498.

Brown, J.S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. Educational Researcher, 18(1), 32-41.

Carroll, J. S. (1980). Analyzing decision behavior: The magician's audience. In Thomas S. Wallsten Cognitive processes and choice in decision behavior (pp. 69-76). Hillsdale, NJ: Erlbaum.

Castellan, N. J. (1993). Evaluating information technology in teaching and learning. Behavior Research Methods, Instruments, & Computers, 25, 233-237.

Chase, W. G., & Simon, H. A. (1973). Perception in chess. Cognitive Psychology, 4, 55-81.

Chi, T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. Cognitive Science, 5, 121-152.

Cohen, S., Tsai, F., & Chechile, R. (1995). A model for assessing student interaction with educational software. Behavior, Research Methods, Instruments, & Computers, 27, 251-256.

Cohen, S., Chechile, R., Smith, G., Tsai, F., & Burns, G. (1994). A method of evaluating the effectiveness of educational software. Behavior Research Methods, Instruments, & Computers, 26, 236-241.

Crocker, J. (1981). Judgment of covariation by social perceivers. Psychological Bulletin, 90, 272-292.

Dawes, R. M. & Corrigan, B. (1974). Linear models and decision making. Psychological Bulletin, 81, 95-106.

Dudley-Marling, C., & Owston, R. D. (1987). The state of educational software: A criterion-based evaluation. Educational Technology, 27, 25-29.

Duncan, N. C. (1993). Evaluation of instructional software: Design considerations and recommendations. Behavior Research Methods, Instruments, & Computers, 25, 223-227.

Edmonds, G. S., Branch, R. C., & Mukherjee, P. (1994). A conceptual framework for comparing instructional design models. Educational Technology and Research Development, 42, 55-72.

Educational Products Information Exchange Institute (EPIE) (1983). Microcomputer courseware evaluation form. Hampton Bays, NY: EPIE.

Educational Products Information Exchange Institute (EPIE). (1994). The educational software selector (TESS)[CD-ROM]. Hampton Bays, NY: EPIE [Producer and Distributor].

Einhorn, H. J. (1980). Learning from experience and suboptimal rules in decision making. In T. S. Wallsten

Cognitive processes and choice in decision behavior (pp. 1-20). Hillsdale, NJ: Erlbaum.

Ericsson, K. A., & Simon, H. A. (1993). Protocol Analysis. Massachusettes: MIT Press.

Gagne, R. M. (1966). Human problem solving: Internal and external events. In B. Kleinmuntz, (Ed.), Problem solving: Research, method, and theory (pp. 128-148). New York: Wiley.

Glaser, R. (1984). Education and thinking: The role of knowledge. American Psychologist, 2, 93-104.

Glaser, R. & Chi, M. T. H. (1988). Overview. In M. T. H. Chi, R. Glaser, & M. J. Farr (Eds.), The nature of expertise (pp. xv-xxvviii). Hillsdale, NJ: Erlbaum.

Gonce-Winder, F. & Walbesser, H. H. (1987). Toward quality software. Contemporary Educational Psychology, 12, 261-268.

Greeno, J. G. (1976). Indefinite goals in well-structured problems. Psychological Review, 6, 479-491.

Greeno, J. G. (1977). Process of understanding in problem solving. In N. J. Castellan, Jr., D. B. Pisoni, & G. R. Potts (Eds.), Cognitive Theory (Vol 3) (pp. 43-83). Hilldale, NJ: Erlbaum. Hayes-Roth, B. & Hayes-Roth, F. (1979). A cognitive model of planning. Cognitive Science, 3, 275-310.

Howat, K., Michaud, C., & Fetzer, D. (1995). Supersolvers OutNumbered! [computer software]. Fremont, CA: The Learning Company.

Johnson, E. J. (1988). Expertise and decision under uncertainty: Performance and process. In M. T. H. Chi, R. Glaser, & M. J. Farr (Eds.), The Nature of Expertise (pp. 209-228). Hillsdale, NJ: Erlbaum.

Kagan, D. M. (1992). Professional growth among preservice and beginning teachers. Review of Educational Research, 62, 129-169.

Lawrence, J. (1988). Expertise on the bench: Modeling magistrates' judicial decision making. In M. T.

H. Chi, R. Glaser, & M. J. Farr (Eds.), The Nature of Expertise (pp. 229-259). Hillsdale, NJ: Erlbaum.

Larkin, J., McDermott, J., Simon, D. P., & Simon, H. (1980). Expert and novice performance in solving physics problems. Science, 208, 1335-1342.

Leinhardt, G. (1986). On teaching. In R. Glaser, (ed.) Advances in Instructional Psychology Vol. 4 (pp. 1-54). Hillsdale, NJ: Erlbaum.

Leinhard, G. & Greeno, J. G. (1986). The cognitive skill of teaching. Journal of Educational Psychology, 2, 75-95.

McDougal, A. & Squires, D. (1995). A critical examination of the checklist approach in software selection. Journal of Educational Computing Research, 12, 263-274.

Newell, A., & Simon, H. A. (1972). Human Problem Solving. Englewood Cliffs, NJ: Prentice Hall.

Payne, J. W. (1980). Information processing theory: Some concepts and methods applied to decision research. In T. S. Wallsten Cognitive processes and choice in decision behavior (pp. 95-115). Hilldale, NJ: Erlbaum.

Ransdell, S. (1993). Educational software evaluation research: Balancing internal, external, and ecological validity. Behavior Research Methods, Instruments, and Computers, 25, 228-232.

Reitman, W. (1965). Cognition and thought: An information processing approach. New York: Wiley.

Roblyer, M. D., Edwards, J.,& Havriluk, M. A. (1997). Integrating educational technology into education. New Jersey: Prentice Hall.

Sabers, D. S., Cushing, K. S., & Berliner, D. C. (1991). Differences among teachers in a task characterized by simultaneity, multidimensionality, and immediacy. American Educational Research Journal, 28, 63-88.

Schlichting, M. (1995). Math Workshop [Computer software]. Novato, CA: Broderbund Software.

Schueckler, L. M. & Shuell, T. (1989). A comparison of software evaluation forms and reviews. Journal of Educational Computing Research, 5, 17-33.

Shavelson, R., & Stern, P. (1981). Research on teachers' pedagogical thoughts, judgments, decisions, and behavior. Review of Educational Research, 51, 455-498.

Shuell, T. J. & Schueckler, L. M. (1989). Toward evaluating software according to principles of learning and teaching. Journal of Educational Computing Research, 5, 153-149.

Shulman, L. S. (1986). Paradigms and research programs in the study of teaching: A contemporary perspective. In M. C. Wittrock (Ed.), Handbook of Research on Teaching (pp. 3-36). New York: Macmillian.

Shulman, L. S., & Elstein, A. S. (1975). Studies of problem solving, judgment, and decision making: Implications for educational research. In F. N. Kerlinger (Ed.), Review of Research in Education, (pp. 3-42). Itasca, IL: Peacock.

Simon, H. (1975). The functional equivalence of problem solving skills. Cognitive Psychology, 7, 268-288.

Swanson, H. L., O'Connor, J. E., & Cooney, J. B. (1990). An information processing analysis of expert and novice teachers' problem solving. American Educational Research Journal, 27, 533-556.

Voss, J. F. & Post, T. A. (1988). On the solving of ill-structured problems. In M. T. H. Chi, R. Glaser, & M. J. Farr (Eds.), The Nature of Expertise (pp. 261-285). Hillsdale, NJ: Erlbaum.

Wineburg, S. S. (1991). Historical problem solving: A study of the cognitive processes used in the evaluation of documentary and pictorial evidence. Journal of Educational Psychology, 83, 73-87.