During the past 20 years, and particularly during the last decade, interest in the computer as an instructional technology has increased dramatically. Virtually all phases of education and training have been affected by the growth of interest in the computer, as evidenced by the proliferation of computer-based instruction products that have come on the market and the popularity of computer-based training in business and government. The computer appears to be a technology of long-term consequence to the instructional development profession.

In planning this issue, the goal was to present a series of papers pertaining to important issues in the design of computer-assisted instruction (CAI). To accomplish this goal, authors from academic and applied settings were asked to develop their ideas concerning CAI within the design focus of JID. The result of their efforts is this special issue, which includes comprehensive summaries of CAI research and the characteristics of an empirically derived CAI system (Tennyson), an evaluation of factors to be considered in making instructional design decisions (Hannafin), an analysis of the underlying assumptions and capabilities of various authoring options (Kearsley), an examination of the implications of learning and cognition research for design of CAI (Clark), and the presentation of several practical models for the evaluation of CAI (King and Roblyer).

The topics addressed in this issue should enlighten, broaden, and stimulate thinking about issues in the design of CAI. The papers do more than present information. They challenge the instructional design profession to examine carefully the fruits, fallacies, and risks associated with CAI, and to be thoughtful in our instructional judgments and decisions.

I am extremely grateful to each of the authors for contributing their efforts to this issue.

-Michael Hannafin
Guest Editor
Research on Student Thought Processes During Computer-Based Instruction

Richard E. Clark
Director, Center for Instructional Research, Development and Training (CIRDaT)
University of Southern California
Los Angeles, CA 90007

It wasn't many years ago that a journal with the stature of JID would have refused an article that promised to review research on thinking. Until recently, strong behaviorist biases have greatly influenced decisions about what is acceptable, both in journal articles and in instruction. Generally, behaviorism has been a positive force in instructional development. Most of our tested, robust instructional methods have been derived from behaviorist principles. Those principles reflect an emphasis on research that examines how instructional variables such as reinforcement, feedback, practice, and measureable objectives directly contribute to student achievement. In contrast, research on thought processes examines how instructional presentations influence what students think, believe, and feel and how these thoughts in turn influence achievement. This cognitive research adds another link to the chain of constructs that connect instruction and student cognition. The second link is between student cognition and learning or performance. The distinctive characteristic of cognitive research is the idea that instruction influences achievement through student thought processes. That is, instruction influences thinking and in turn thinking influences learning and performance. The cognitive approach therefore assumes that instruction is mediated by student thought processes.

What follows is a brief sampling of cognitive research in the areas of motivation, computer effects, learner control, instructional methods, left brain, right brain effects, and anxiety. These capsule status reports on each research area are frankly offered to encourage design professionals to consider cognitive views. Since space is limited and these summaries skip over complexities, the reader is encouraged to follow up the citations provided before using these limited generalizations in instructional design. Each section begins with an attempt to summarize the most defensible conclusion of current research and thinking about the area. However, the reader is cautioned that other views exist; compelling alternatives are indicated by citations.

1. The motivation or effort students invest in computer-based instruction depends on, in large measure, on their beliefs about how difficult it is to learn from computers. Beliefs that computers are either very easy or very difficult may result in less persistence at learning tasks.

Behaviorist views of motivation linked increased persistence at a task with reinforcement. It was probably Julian Rotter (1966) who first documented the fact that many human subjects in experiments did not persist when reinforced. He attributed it to an external locus of control of reinforcement and suggested that these externals did not notice reinforcement since they were fatigually unconcerned with the effects of their own learning strategies. External learners believe that chance or fate governed their performance and reinforcement was viewed as arbitrary. Only "internals" persisted with reinforcement, which they take to be evidence for the efficacy (success or failure) of their efforts to learn.

Salomon (1983) has been responsible for applying a cognitive model of motivation to understanding the student thinking process that influences the effort spent on CBI. He has claimed that persistence at instruction depends, in part, on the students' judgement about how difficult it will be to learn from media such as television or computers. The relationship that Salomon proposes between difficulty beliefs and effort invested follows the classical, arousal theory "inverted U" shape. That is, learners will invest relatively less effort when they believe that computers are moderately difficult. Salomon describes this phenomenon in a model which hypothesizes a relationship between variables like PDC (Perceived Demand Characteristics) of the medium) and AIME (Amount of Invested Mental Effort). One could interpret Salomon's model and data to suggest that CBI may fare better than television as an instructional medium since many students attribute difficulty to learning from computers but tend to think of television as "shallow" and easy.

Another sense in which beliefs about computers influence motivation can be found in the literature on the social psychology of instruction. Hess and Tenerezis (in Clark, 1983) reported that Mexican-American, low SES middle school students liked remedial CBI math better than teacher presented math because they believed the computer to be more fair. Presumably, they perceived that the computer gave consistently accurate feedback whereas teachers occasionally were making decisions based on student heritage. The positive attitudes towards CBI translated into increased persistence and helped increase their scores on math tests over other instructional options. This was true even though the same students thought that teachers were more flexible and responsive to student wishes to change the direction or content of instruction.

Clark (1983) has reviewed other attitude and belief studies that suggest a relationship between teacher attitudes towards CBI and their attitudes towards science and mathematics. There is some evidence that, if
teachers dislike science and math, they will also dislike CBI.

It is important to keep in mind that beliefs about computers are more or less arbitrary and changeable. In a research sense, beliefs are learner variables, not instructional variables. If we find a relationship between beliefs and effort, the presumed cause is not the computer but the student’s belief. We therefore cannot conclude for example, that Fless and Tenezakis found out something about the influence of computers on motivation. We may only conclude that the typical relationship between perceived demand and effort may also influence the effort students will spend on CBI.

2. All learning benefits that are attributed to computer-based instruction are readily available from other media. Analysis of computer instruction research indicates that they have no new or unique qualities that cause changes in student learning.

CBI is such a new and compelling instructional tool that most designers are curious about its potential to make a unique contribution to learning and performance. There are also many research and evaluation studies that show large learning gains from computers when CBI is compared to other media such as live teachers, textbooks, television, and audio-tutorial media. What makes the problem confusing is that there are many other studies which just as clearly demonstrate that these other media are superior to computers. Advocates from one or another medium can always find evidence for their point of view but the objective observer has had difficulty siftling among conflicting claims.

Cognitively oriented researchers have viewed the problem from the student’s perspective. What they ask, would be the impact on student cognitive learning processes as a result of the use of computer versus some other medium? In every case, it seems that computer features which might influence cognition are available from some other medium, often easily available. Therefore, it is impossible to speak of unique contributions to learning from computers (cf. Clark & Salmon, in press). Why then do so many studies indicate that CBI leads to greater achievement than other media? Recent meta-analytic studies provide some of the answer.

In meta-analyses of hundreds of CBI studies, there is clear evidence of consistent confounding in the research (Clark, 1983).

It seems that when CBI is found to promote more learning than some other medium (e.g., classroom-based, teacher-centered instruction), closer inspection of the studies show that some other factor probably caused the learning gains. The most powerful other factor that is seldom controlled in these studies is a greater effort to design the CBI presentation than its competitor. This greater effort probably results in more effective instructional methods being used in the CBI presentation. When the same methods and content are employed by all media being compared, there are no significant learning advantages found for CBI.

This lack of a learning gain from CBI use is particularly evident when long-term studies are reviewed. In shorter term observations (three and four weeks) students seem more motivated to invest more effort and learn more because the CBI experience is novel and exciting. As with all former media, however, the novelty wears off after a few weeks and learning benefits diminish.

Learning gains come from adequate instructional design theory and practice, not from the medium used to deliver instruction.

One of the most obvious lessons from this comparative research is that learning gains come from adequate instructional design theory and practice, not from the medium used to deliver instruction. It is also important to note that there may be cost benefits and student enjoyment advantages to using computers for instructional purposes only when they do not contribute to unique learning gains.

There are and will continue to be disputes about the role of computers and other media in directly influencing learning. A particularly interesting expansion of the cognitive point of view on this issue is offered by Salomon and Gardner (1984).

3. When computer based instruction offers “menus” of learner and system controlled study aids, students may learn less from the options they like the most. This may be most true for students at the extreme ends of the ability scale.

One of the advantages of CBI is its potential to easily offer instructional support to students on demand. This “student assigned” (Rigone, 1980) capacity of CBI is used in many contemporary software programs and is a feature suggested to new CBI designers. And yet, the research evidence shows that CBI support menus have very mixed learning consequences. Clark (1982), in a review of aptitude-treatment interaction (ATT) studies on the issue, offered a cognitively-based explanation for the mixed results. He found that high general ability students typically seek high support during learning when they are offered a choice. Higher support often means additional examples, practice exercises, organizational strategies, self-testing, and study strategies. Yet, these same high ability students learned much less with high support than they did when left more on their own. Conversely, lower ability students tended to prefer less support in the form of self-paced, “discovery” instructional options.

Here again it was an analysis of student thinking that provided an insight. It seems that students incorrectly assess the impact a particular kind of instructional support will have on their learning. High ability students tend to use supports quite often even though they typically achieve more when they use their own skills. Clark hypothesized that able students believe that more support will make their effort to learn more efficient. On the other hand, low ability students tend not to use supports when given the choice. Clark suggests that these students are avoiding the extra effort provided by high support options when they expect to fall anyway. Lower support methods usually allow the student to loaf.

It is important to note that this finding was only possible when ATT researchers asked how instructional treatments were mediated by student
ability. Many CBI evaluations do not include ability differences and therefore can report only that the evidence linking support menus and achievement is mixed.

4. There may be very different transfer consequences produced by designs which follow behaviorist models versus those which are cognitively based. Behaviorist methods seem to support context-bound learning whereas learning from cognitive methods may be more generalizable.

Though the computer will accommodate nearly any instructional design model, designers typically assume that the medium is more amenable to behaviorally-based design. This is probably due to early attempts to adapt programmed instruction methods in the first demonstrations of computer assisted instruction. Whatever the reason, there is compelling recent evidence (e.g., DiVesta & Feverly, 1984) that behavioral designs promote different types and generalizability levels of achievement than do cognitive designs.

Highly directed instruction produces the kind of behaviorally-based learning often desired by military and business clients.

When designs specify the use of highly directed, shorter step instruction with more feedback, with opportunities for specific practice, and with identical elements in both lesson and test environments, learning tends to be more “near transfer.” Near transfer means more specific to the examples and context where it was learned and therefore more difficult to use to solve novel problems. This is the kind of behaviorally-based learning often desired by military and business clients.

On the other hand, when design models specify student control over the direction and monitoring of instruction at higher difficulty levels and in larger chunks, learning tends to be available for “far” transfer. Here skills are more easily generalized to novel settings or problems. This is the design model often favored by clients who need to train students to solve novel problems in settings that are difficult to specify in advance (e.g., formal schooling, management problem solving, command decisions). These design characteristics are more typical of cognitive instructional design models (e.g., Reigeluth, 1983, for examples of both types of design).

It is important to note that near transfer may be achieved somewhat at the expense of far transfer (and vice versa). There is evidence from a number of recent studies that task-specific skills are gained initially at the expense of ability to transfer those skills to new contexts. The same seems to be true for the more generalizable, far transfer learning—it is simply not as efficient as near transfer learning in solving any specific application problem. This might provide some of the more rational basis for the disputes between proponents of these two kinds of design models (cf. Clark & Voogel, 1984).

This distinction is important to CBI design because many designers assume, without examination, that CBI must utilize behavioral models. Of course, this is not the case, even though it has been common practice in the past.

5. Hemispheric dominance and brain lateralization studies will probably not yield important CBI design prescriptions. The preference of each hemisphere seems to be fleeting, influenced by expectations, and not obviously influential in complex learning.

Instructional designers occasionally wonder whether research that indicates information processing differences between the left and right hemispheres can be adopted to CBI design. Essentially, there is research evidence that the left cerebral hemisphere specializes in more verbal and analytical messages and that the right hemisphere favors spatial and aesthetic information. This is true for right-handed learners; the reverse is the case for left handed learners. This research area has been the impetus for a number of popular articles and workshops focused on methods of “training the right brain.” These popular presentations are only loosely based on the available research (cf. Gardner, 1982).

In a review of both psychobiological and instructional research on the topic, Hellige (1980) cautions against attempts to apply this research to design models. There has been little in the research since his review to contradict his caution. Hellige’s argument is that in the normal brain, both hemispheres are strongly and able to perform each other’s functions. The main difference between them seems to be a momentary advantage in the decoding of different types of information. However, he notes that the hemisphere initially activated will depend as much on our expectations for the kind of information we are going to receive as on the exact type we do encounter.

6. Efforts to alleviate the kind of test anxiety that inhibits performance in CBI and testing have largely failed. Many anxiety researchers now assume that instructional anxiety may have to be circumvented rather than cured.

Highly anxious students tend to learn less and drop out of CBI programs more frequently than those who experience only moderate anxiety. Anxiety-reducing elements in CBI design would be a positive contribution to learning and performance. However, there has been a recent reformulation of theory concerning the way that anxiety influences learning which may significantly change our design strategies in this area (Tobias, 1983).

Whereas in the past we have tended to use desensitization therapy procedures to decrease anxiety, there is evidence that this method only reduces the emotion connected with fears; the learning itself is still depressed after desensitization (cf. Tobias, 1983). This implies that our current methods for decreasing computer related anxiety will influence only its non-learning elements. Recent explanations for this finding point to two possible contributors: interference and skill deficits.

The interference explanation for the influence of anxiety on learning notes that fear of failure may be two-sided: on the one hand anxiety produces psychological responses such as shaking and sweating, and on the
other hand there are cognitive (worry) consequences. It is the physiological responses that are alleviated by desensitization, but worry seems to be left intact. Worry (the compulsive thinking about fears) is said to interfere with the mental processing of 'space' needed for learning, and thus achievement is reduced (Tobias, 1979).

The second contributor to decreased learning is thought to be skill deficits. Analyses of student thought processing during learning suggest that students who assume that they lack the skill necessary to successfully complete the CBI lesson will experience anxiety (Tobias, 1983). Notice that anxiety is a dependent measure in this research. Here, researchers do not assume that anxiety causes learning problems but instead that student conclusions about their own skills produces anxiety. It is the skill deficit that lowers learning scores.

Since worry seems difficult to treat, it is possible that the best we can do now to overcome the effects of stress in CBI is to increase the level of instructional support. If interference is contributing to lower learning scores, more support for the student will lower the demand on processing space. If a lack of skill is the problem, increased support will tend to help also, at least over the short term. Tobias recommends ample opportunity to review instructional material, shorter steps, a high degree of embedded organization, the use of good examples, and constant feedback on progress. Note again that these methods do not decrease worry or anxiety, they tend to circumvent it by reducing the cognitive load on the anxious (worrying) learner.

Conclusions

The purpose of this article is to illustrate the distinctive perspective that underlies research on student thought processes during instruction. In this perspective, it is assumed that learning results from instruction only after that instruction has been considered by students. These considerations or cognitions are thought to mediate the effects of instruction and, in turn, influence learning, performance, and transfer. The cognitive approach to research on instruction is offered as a supplement to behaviorist strategies, not as a replacement. It is highly likely that some learning occurs with minimal cognitive mediation and behaviorist instructional methods are recommended as ideal when near transfer (context bound) learning is desired. Yet, it is also the case that the cognitive view of learning provides significant alternative views of certain behaviorist-inspired instructional models. Evidence was presented for cognitively-inspired alterations in our understanding of such familiar instructional variables as motivation, the relationship between CBI and learning, learner control, the transfer of learning, and hemispheric dominance, and anxiety. In each case, an analysis of student thought processes have extended the robustness of the principles available to support instructional theory and design.

References


Hemispheric dominance and brain lateralization studies will probably not yield important CBI design prescriptions.

Guidelines for Using Locus of Instructional Control in the Design of Computer-Assisted Instruction

Michael J. Hannafin
Instructional Systems Program
College of Education
The Pennsylvania State University
University Park, PA 16802

Abstract. Computers offer a variety of instructional control options to designers of computer-assisted instruction. However, the amount and type of instructional control is affected by both the nature of the learning task and learner characteristics. The purposes of this paper are to present empirical evidence on locus of instructional control, and to present guidelines for determining learner versus lesson control in computer-assisted instruction.

The computer as an instructional delivery system offers the designer a myriad of options. Instruction can be designed using a variety of presentation formats, interaction options, feedback techniques, and instructional management options. Computers, as a consequence, have been praised for their capabilities and the power offered as an instructional technology. Knowledge of how and when to use these capabilities, however, has been considerably slower to develop.

One of the most powerful and important features of the computer is the virtually unlimited range of instructional control options available to designers of computer-assisted instruction (CAI) (Burke, 1982). Although substantial research has been conducted related to instructional control and CAI, little of this information has affected the instructional design and development profession. A variety of factors affect the decision of how much, or what type of, instructional control is desirable. The purposes of this paper are to examine several factors that should be considered, and to present a set of guidelines for determining the instructional locus of control in CAI.

Instructional Locus of Control and Computer-Assisted Instruction

Locus of control as a psychological construct has been studied extensively. Instructionally, however, locus of control has assumed a different meaning. Typically, instructional locus of control has been examined by manipulating instructional features such as method of lesson pacing (Ross & Rakow, 1981), management and evaluation decisions in instruction (Hannafin, 1981), en route decisions regarding need for additional instruction (Tennyson, 1981), and selection of other instructional features (Carrier, Davidson, Higson, & Williams, 1984). Instructional locus of control can be thought of as a continuum ranging from fully externally controlled to completely internally controlled. Instruction is considered to be more externally controlled with fixed rate, linear delivery systems such as slide-tape presentations; instruction is thought to be more internally regulated in delivery systems where the learner exercises significant control over the contingencies of the lesson, such as in certain CAI lessons.

In this paper, external locus of control is defined as instruction in which all learners follow a predetermined path established by the designer without exercising individual judgement as to the appropriateness of the path. Internal locus of control is demonstrated in lessons where individuals control the path, pace, and/or contingencies of the instruction, typically by specifying choices among a range of designer-embedded options. While combinations of internal and external control are perhaps most common, this paper will focus on each type separately.

Empirical Perspectives:
External vs. Internal Control

External control does not necessarily connote linearity. While linear CAI designs are an instance of externally controlled instruction, externally controlled lessons more typically offer a variety of branching options—all of which are executed under fixed lesson rather than learner control. Several researchers have developed successful externally controlled adaptive strategies (Park & Tennyson, 1983; Ross & Rakow, 1980, 1982; Ross, Rakow, & Bush, 1980; Rothen & Tennyson, 1978; Tennyson, Christenson, & Park, 1984; Tennyson & Rothen, 1979). In such designs, contingencies are typically established which control the need for, or selection of, systematic branching. One student may learn all intended information rapidly, and never branch for review or remediation. A different learner, however, may experience difficulty throughout the same lesson, and the lesson will adapt and branch the learner to appropriate lesson segments as needed. In each case, movement through the lesson is dictated by the designer, and presumably by an overriding justification for routing learners through prespecified paths. Such strategies have proven effective for teaching a variety of skills and for reducing instructional time (Kulik, Kulik, & Cohen, 1980; Edwards, Norton, Taylor, Weiss, & Dusseldorp, 1975).

One of the principal criticisms of such designs, however, is the tacit assumption that the designer is the best judge of when, where, and how much instruction is needed to learn a given skill. Since learners cannot control directly the instructional sequence, frequency of examples, or number of practice items, faster learners may be...
required to complete instructional sequences that are unnecessary or inappropriate given their individual learning styles. A wide range of internal or learner control, strategies have also been developed and studied (Bunderson, 1974; Caffarella, Caver, Legum, Shtogren, & Wager, 1980). In many cases, however, learner control in CAI has proven less successful (Steinberg, 1977). Despite the inconsistencies, Snow (1980) has argued that while performance has rarely been optimized under learner control in the past, the conditions of effective learner control still warrant study.

To this end, a variety of "coaching" procedures have been studied (see, for example, the extensive study reported by Ross and Rakow and their associates, and Tennyson and his associates). Researchers have successfully developed procedures that offer guidance upon which individual learner's decisions can be based. Learners may be advised as to the number of practice items or examples recommended, based upon the individual learner's past, current, or cumulative performance, during a lesson. However, the learner maintains control over the instructional decisions by accepting or rejecting the advice offered during the lesson, and proceeding as individually deemed appropriate. In effect, learners make informed judgement regarding their instructional sequence, as opposed to making an uninform ed decision.

Other Findings
A number of additional patterns have emerged from the study of instructional control in CAI. Externally controlled CAI has proven effective in a variety of drill and practice tasks (Kulik, Bangert, & Williams, 1983; Merrill & Salisbury, 1984; Sarach, 1982). This may result since such lessons typically reinforce previously taught information rather than teach new instructional content. In cases where standards for mastery are already established, external control can force learners through the mandated number of practice trials and require learners to demonstrate desired levels of mastery during the lesson, thereby ensuring the quality of performance. It is unclear, however, whether or not externally controlled CAI is necessary for learning under such circumstances, or if learner controlled instruction could prove more effective or efficient than externally controlled instruction. It is possible that a significant amount of control could be transferred to the learner with equal or greater success.

Although the findings are inconsistent, learner age and ability have also been found to affect the extent to which learner control strategies can be effectively applied. Whereas most CAI studies have been conducted using college students, who are older and generally capable academically, recent studies suggest that younger and less able learners may not perform well under internal instructional control (Goetzfried & Hannafin, in press). Older and more able learners may have more effective and refined cognitive strategies to apply during instruction, and are likely to be better at estimating the accuracy of learning, the presence of confusion, and the need for additional instruction than younger and less able learners. In effect, providing the option for control to older and more able learners may enable them to apply individually developed cognitive strategies to sort and assimilate information in ways that are uniquely effective. Younger and less able students, on the other hand, may have neither the refined cognitive strategies nor the self-evaluation skills to apply during a lesson. For these reasons, structured, externally controlled CAI may provide a superior organization of to-be-learned information, while eliminating the need for self-evaluation, for younger and less able learners.

Internally controlled lessons are generally more time-consuming and consequently more costly to design and to develop than externally controlled lessons.

Practical Concerns
One of the most important considerations in the instructional control decision may be the ease, or difficulty, of design and development of internally versus externally controlled CAI. Internally controlled lessons are generally more time-consuming, and consequently more costly, to design and to develop than externally controlled lessons. This is due largely to the extent to which the designer must anticipate a range of learner options, each with a corresponding set of unique response contingencies. Internally controlled CAI designs tend to require a more complex set of branching options than externally controlled designs where instructional contingencies are typically established across learners.

The effect of choice on learner attitudes has also received attention. It has been suggested that learner control may be related more to learner attitude than to achievement (Dalton & Hannafin, in press; Hannafin, 1982). Researchers have reported positive effects of different CAI lessons on attitudes toward both the information studied and the computer itself (Fowler, 1983). The learning experience offered by the computer (Lawton & Gerschner, 1982), and, in some cases, compared with competing delivery systems (cf. Kulik, 1983). However, the comparative effects of different CAI control options on attitudes have not been studied extensively. This is a particularly interesting and important issue, since the computer offers the potential to provide as much, or as little, control to learners as appropriate and desirable. Presently, there is insufficient empirical evidence to support the comparative superiority of internally, or externally, controlled CAI on learner attitude.

Two related issues, studied less often but of potential importance, are the extent to which learners select paths that are different from designer imposed paths, and the extent to which such choices affect learning. Nested within these issues are several fundamental questions concerning instructional locus of control in CAI. If, for example,
Tentative Guidelines for Determining the Locus of Instructional Control in CAI

Although the answers to the preceding, and other, questions have not been conclusively demonstrated with CAI, a number of tentative guidelines may be proposed. The following guidelines are based upon research in the instructional design, learning and cognition, and CAI fields, as well as the author's experience. The guidelines pertain to learner age and ability, the nature of the learning task, the use of coaching procedures, the inclusion of structural guidance, and procedures for monitoring lesson implementation.

1. Older students perform more effectively under guided learner control; younger students perform best under lesson control (cf., Fischer, Blackwell, Garcia, & Green, 1975 vs. Goetzfried & Hannafin, in press). The overwhelming majority of the published experimental research on control of instruction has been conducted with older learners (see, for example, the studies by Tennyson and associates and Ross and associates). While internal strategies, especially those using some form of coaching, have been successful with college-age learners, there is simply an inadequate basis for generalizing these findings to younger learners. Internal control by younger students may eventually be demonstrated, either through certain lesson options and structure, or by training individuals in making control decisions, a stronger case can be presently advanced supporting external control for younger learners.

2. More able students perform best under learner control; less capable students perform best under lesson control (Goetzfried & Hannafin, in press). Given the opportunity to apply well-cultivated strategies to learning from the computer, more able learners appear capable of processing information in uniquely efficient and effective ways. The use of computer guidance, for example, tends to be more effective than for less able learners. Less able learners, on the other hand, are likely to profit more from the instructional presentation logic of a knowledgeable designer. Such students may be less effective at evaluating the learning process, and consequently may be inadequate at selecting practice, examples, and the need for reinstallation.

3. Locus of instructional control is dependent upon the nature of the learning task (cf. Gagne & Briggs, 1979):
   a. Procedural tasks are best taught using lesson control. When a sequence of steps or tasks must be learned, the order among the steps must be controlled. Whereas some learners may learn procedural tasks effectively under internally controlled instruction, a greater proportion of learners will likely profit from an established instructional sequence.
   b. Verbatim learning tasks are best taught using lesson control. Where verbal information of a verbatim or literal nature is to be learned, the need for control over the exactness of the presentation increases. External strategies provide a greater degree of certainty of exactly what has been learned, since it is possible to validate learning through mandatory skill checks.
   c. Contextual and substantive information are best taught using learner control. Though not established conclusively for CAI, internal strategies may yield greater depth of processing of presented information. Internal designs permit the learner to form individually relevant associations among prior and current information, thereby deepening and enriching the level at which instruction is processed.
   d. Lower-order intellectual skills are best taught using lesson control; higher order skills may be best taught using learner control. In general, lower order intellectual skills tend to be readily and uniformly classified. Considerable agreement exists, for example, for simple discriminations, concrete concepts, and rules. Such skills may be addressed very efficiently by presenting the basic bodies of instruction to learners under imposed program control. On the other hand, higher order intellectual skills, such as those involving problem solving, require a greater level of abstraction, integration, and application of information. As such these skills may be more amenable to the application of individual judgements as to when and what type of instruction is needed.
   e. Lesson control is desirable for learning tasks with established performance of mastery criteria; for tasks that do not have specified mastery criteria, imposed lesson control is useful for tutorial, and internal control useful for drill and practice. The ability to establish mastery of instructional objectives makes external control desirable under many circumstances. When no fixed mastery criteria exist, however, there is little to lose and potentially a great deal to gain in increasing learner control.
   f. Imposed lesson control is more effective for unfamiliar learning tasks, and learner control more effective for familiar learning tasks (Ross & Rakow, 1981; Tobias, 1982). The degree to which learners have familiarity with to-be-learned information affects the accuracy with which they can make informed control decisions. As familiarity decreases, the amount of structure and explicitness of the instruction needs to be increased. As learners become increasingly familiar with the information to be presented, they appear to become more adept at evaluating their performance, and can make more effective choices.

Older students perform more effectively under guided learner control; younger students perform best under lesson control.

g. Students who perceive themselves as internally governed, i.e., assume personal responsibility for their performance and behavior, perform best under internally controlled CAI; students who perceive themselves as externally governed, i.e., respond to imposed instructional demands, respond best to externally controlled CAI (see Cartier et al., 1984; Holloway, 1978). Although the findings are not definitive, the orientation of the learner, that is the extent to which the learner believes that s/he is affected more by external versus internal events, may be a useful consideration in determining who might be a receptive candidate for transfer of instructional control. If a learner is already oriented to assuming per-
sonal responsibility for their learning, internal control techniques might be an effective instructional technique. Students who are not oriented this way, however, may experience greater difficulty when required to monitor their learning, make decisions regarding options, and otherwise adapt to a learner control procedure.

4. Internal control strategies should include some form of coaching to assist learners in making informed decisions (Ross, 1984; Tennyson & Buttery, 1980). Learner control is not compromised by coaching on such topics as cumulative performance, comparisons to expected standards, recommendations for examples, and a variety of other topics. To the contrary, failure to provide meaningful guidance can prove frustrating to learners in that they may be unable to make intelligent, informed choices.

5. Internally controlled lessons should include a "catch net" to identify ineffective learners. It cannot become not only ineffective, but also frustrating, for students who experience difficulty in attempting to proceed through a learner-controlled lesson indefinitely. In the author's experience, the inclusion of error detection procedures, designed to make alert the student and the instructor to the problem learners, are desirable. As a fail-safe, learners who are struggling during internally controlled lessons should be identified, advised in strategies to improve their judgements, or counseled to consider externally imposed options.

Instructional technologists have been accused of promoting hardware rather than intelligent and informed instructional applications.

6. Internally controlled lessons should include the "exemplar path" normally prescribed by the designer for external control. Assuming a designer has identified a path deemed to be most effective, common sense alone dictates that internally controlled lessons provide at least the same option. It would appear senseless to exclude the instructional sequence and activities advocated by a professional designer in order to defer entirely to an open-ended learning sequence. Learners should be advised as to which options are thought to be most effective at each decision point.

7. Structural guidance, such as provided through the Events of Instruction (Gagne, Wager, & Rojas, 1981), should be provided in both internal and external control designs. Regardless of the CAI control procedures adopted, rational, time-tested instructional components should be included. Learners in internally controlled lessons should be encouraged to participate in key lesson components, such as practice and feedback, in order to improve the probability of learning intended information.

8. Conventions in the use of locus of instructional control should be consistent; changes in control procedures should be explained to the learner. Uniform protocol in CAI is important not only to screen-face design (Haines, 1964), but also to response expectations (Hannafin & Peck, in press). Learners should develop consistent expectations of the control orientation of instruction. If changes in procedures are made, learners should be advised in order to ward off possible frustration.

9. If initial learning is ineffective under one control strategy, switch to the other control option (if available). In the author's experience, an initial, but unsuccessful, pass through a lesson often helps to orient the student to review or remediation. Learners who interact with externally controlled lessons are likely to have learned from their initial exposure to the information. In such cases, they may be capable of making very well-informed choices as to where additional instruction is needed. Correspondingly, learners who have experienced difficulty

To be more consistent in their use of examples, practice, and repetition than ineffective and less able learners. However, the effectiveness, and necessity, of such options for high achievers has been questioned (Clark, 1982). Recording the control choices of learners will help to identify unnecessary options, as well as those options that are most frequently used by effective versus ineffective learners. It is easier to evaluate and to revise existing lessons, as well as to plan future lesson options, based upon such information.

Closing Comments

These guidelines should provide a basis for further study. Certainly, the guidelines require validation across a variety of settings and subject areas. It is also possible, as suggested in related instructional research (Hannafin, 1981), that locus of control effects may be related to factors other than achievement. For example, the effects of learner versus lesson control on non-achievement measures such as attitude, persistence, and continuing motivation, should be addressed.

As an instructional medium, the computer offers options that are tempting, but perhaps unnecessary and even contraindicated in many cases. The importance of harnessing the instructional potential of the computer cannot be overstated. Instructional technologists have been accused of promoting hardware rather than intelligent and informed instructional applications. The emergence of the computer, and the range of its instructional capabilities, makes the computer a unique challenge. Consideration of the proposed guidelines, while offering no guarantee of effectiveness, will provide a start in assisting designers to apply the capabilities of the computer in a well-informed, systematic manner.

References

Carter, C., Davi G., Hisson, V., & Williams, M. (1984). Selection of options by field independent and field dependent students in a computer con-


Instructional Design and Authoring Software

Greg Kearsley
Courseware, Inc.
20075 Carroll Canyon Road
San Diego, CA 92131

Abstract. This article explores the instructional design aspects of authoring software used to develop Computer Based Instruction (CBI) programs. The significance of following good design principles for interactive instruction is emphasized. The nature of authoring languages and authoring systems is discussed. A series of practical considerations in using authoring software and implications for instructional design are described. Finally, needed developments in authoring software including intelligent tutoring and automated instructional design systems are outlined.

With the increasing use of computer based instruction for a wide variety of education and training applications, a great deal of attention is now being focused on the design and development of instructional software (e.g., see Kearsley, 1984; Walker & Hess, 1984). One area of particular attention is authoring software, i.e., programs which help authors create computer based instruction in a timely and cost-effective manner. The relationship between instructional design and authoring software is explored in this article.

Design of Interactive Instruction

Before discussing the characteristics of authoring software, some remarks about the design of interactive instruction are in order. Since a number of good texts have now been written on this topic (e.g., Allessi & Trollip, 1985; Dennis & Kansky, 1984; Steinberg, 1984) extensive discussion is unnecessary here. However, it is critical to understand that such design principles exist and directly affect the quality and effectiveness of instructional programs. For example, consider the ingredients of good screen design (e.g., Grimm, 1983; Heines, 1983; Jenkin, 1981). Good screens are not crowded, avoid full screen scrolling, organize information functionally, are titled, use color and graphics effectively, focus the attention of the learner, and use windows or screen partitioning to present concurrent information.

Conversely, poorly designed screens are too crammed with information, scroll, lack titles, have information located randomly, do not use color or graphics, fail to focus attention, and attempt to present concurrent information in a sequential fashion.

The creation of good screen displays depends upon knowledge and understanding of these principles, not the particular authoring software used. Quality CBI results from following good design principles, not the use of particular authoring software. While this point may seem obvious, a great deal of time is sometimes spent debating the best way to implement an instructional program while neglecting the design principles that really make the difference.

This is especially true of the pedagogical frameworks that usually underly the development of authoring programs. For example, the TICCIT System was based upon a model of componentized learner control that characterized all instruction developed for that system (Bunderson, 1974). Gagne, Wager and Kojas (1981) outline a model for developing programs based upon Gagne's instructional theory. Bork (1984) describes a methodology which prescribes different contributions made by various team members. These design and development frameworks affect the nature of the instructional programs produced much more than the particular authoring software used.

Authoring Computer Based Instruction (CBI)

The range of authoring software available today, is quite impressive. Historically, authoring programs were developed for dedicated CBI systems (such as PLATO and TICCIT) or IBM mainframe computers (such as IIS, Phoenix, or Scholar/Teach). The evolution and characteristics of these early authoring systems are discussed in Kearsley (1982). With the emergence of microcomputers, dozens of authoring programs for popular computers have become commercially available (see Kearsley, 1984). Furthermore, a number of special purpose authoring programs (e.g., for creating graphics, interactive video or simulations) have also appeared.

Authoring Languages and Authoring Systems

An important distinction necessary to understand authoring software is the difference between an authoring system and an authoring language. An authoring language is a special purpose programming language designed solely for creating instructional programs. An authoring system is a high level interface that allows an author to generate an instructional program without any explicit programming, simply by specifying the instructional content and teaching logic. The distinction between an authoring language and an authoring system is important because it reveals a fundamental characteristic of authoring software. Both authoring language and authoring systems provide structure to the instructional design process. An authoring language provides simplified programming commands for creating screen displays and graphics, for answer analysis, and for branching. An authoring system goes one step further and provides an actual lesson framework, complete with an implicit or explicit pedagogical strategy. It is this structure of an authoring language or authoring
system that makes it possible to create instructional programs faster and with less effort than using a general purpose programming language (such as BASIC or Pascal).

However, it is also the same structure that imposes the major limitations of authoring software. As long as the pedagogy built into the authoring language or authoring system is appropriate to the application (or acceptable to the author), then the tool is useful. But, if the structure limits what the author would like to do, the authoring software becomes a problem and constrains the development process and the quality of the instructional programs created. Hence, the dilemma for the developers of authoring software: how to design authoring tools that provide maximum flexibility in the creation of instructional programs, while also providing sufficient structure to minimize the time and effort required in the authoring process.

**Authoring Languages**

There are many authoring languages available for both dedicated CBI systems (e.g., PLATO, TICCIT) and personal computers. Three authoring languages in particular have strongly influenced the CBI field: Coursewriter, Tutor, and Pilot. Coursewriter was originally developed by IBM for their 1500 Instructional System and then became the basis for the Interactive Instructional System (IIS) which runs on IBM mainframe computers. IIS is widely used in the training world, especially for applications involving data processing or administrative training.

Tutor is the authoring language developed for the PLATO system. Tutor is probably the most complex authoring language in existence with hundreds of commands and parameters. It has been used to create thousands of PLATO programs for a diverse range of educational and training applications. A modern derivative called TenCore is now available for IBM personal computers.

Pilot was specifically designed as a simple authoring language that could be used by teachers. One of the things that makes Pilot easy to learn is a relatively small number of commands (about 20). Pilot has the distinction of running on most microcomputers and hence comes closest to allowing transportability of instructional programs from one computer to another. Because Pilot is available for most microcomputers, it is also the best known but not the most used authoring language. All three of these authoring languages exhibit similar features. This includes commands to:

- display text on the screen
- match student responses
- assign values to variables
- branch instruction on specified conditions
- collect student performance data

The three languages differ in their capabilities to create graphics, character sets, sound, and handle non-keyboard input (e.g., touch, joystick, light pens).

Authoring languages such as Coursewriter, Tutor, or Pilot differ from general purpose programming languages (such as BASIC, Pascal, Fortran, etc.) mainly in the additional capabilities provided for creating screen displays, answer analysis, and collection of student performance data. The net result of these differences is that authoring languages make it faster to create an instructional program since the features necessary or desired for instruction are present in the language.

Authoring languages provide relatively few constraints on instructional strategies or pedagogy, unlike authoring systems. For example, you can program tutorials, drills, games, simulations or tests using any authoring language. Depending upon the features of the specific authoring language, it may be easier or harder to do what you want—but it is undoubtedly possible given sufficient programming expertise. The last remark reveals the real limiting factor with authoring languages; they are as good as the programming skills available. The development of high quality instructional programs using authoring languages is not only dependent upon good design but also, upon the availability of very experienced programmers. Should instructional designers attempt to do the programming themselves, they will need to become accomplished programmers in order to implement their designs in a sophisticated fashion.

**Authoring System**

An authoring system is basically a program that generates other programs. By choosing options from menus or responding to questions, the information needed to create an instructional program is provided without actually having to write the program. This means that an author does not have to learn to program or even rely on a programmer to produce lessons. The lesson content can be entered in the computer directly from storyboards or scripts without having to be translated into programming instructions.

When the system generates an instructional program, the program is free of "bugs." Since debugging a program typically account for about 50 percent of the total time spent developing software, the elimination of the debugging step is a tremendous improvement in productivity. Thus, the two major advantages of authoring systems over authoring languages are:

1. Programming skills are not required, and
2. Considerable time is saved because no debugging is needed.

As explained earlier, the primary limitation of authoring systems is that they can significantly constrain what can be done instructionally. For example, suppose that the question (or test item) component of the authoring system allows a wide range of possible question types but you want to create a type of question which is not provided. Or, suppose, you wanted to measure how long it took the student to answer the question and this was not a feature built into the authoring system. Or, perhaps you want to create a game, but there is no scoring capability available. Most instructional designers can very quickly generate ideas that cannot be accommodated by the fixed format of an authoring system.

There are basically three solutions to this dilemma. The first is to accept the limitations of the authoring systems and avoid designing any instruction that can't be done using the authoring system. Keeping in mind that authoring systems can substantially decrease the time and costs associated with developing lessons, this may be a desirable strategy for some applications.

The second solution is to use a combination of authoring systems and author/general purpose programming...
languages. The authoring system is used as much as possible, but things that can't be done using the authoring system are programmed and incorporated into the program via an authoring or general purpose programming language. Some authoring systems include an authoring language and make it easy to go between the system and the language. This strategy allows the author to take advantage of the efficiency of the authoring system as much as possible and to have the flexibility of an authoring language when needed.

The third solution is to allow the author to modify the authoring system and add new instructional features. This is done by designing new templates which include the desired capabilities. For example, if you wanted to have multiple responses to a question, you would use the authoring system to create a new option to do this. In some authoring systems that allow this capability, all authoring programs are just lessons themselves and they can be created or modified in the same manner as any other lesson created.

**Authoring Software Capabilities**

From an instructional design perspective, the most important aspect of authoring software is the range of capabilities provided. The more functions provided, the greater flexibility possible and the wider range of instructional applications which can be created.

Table 1 provides a list of desirable capabilities for an authoring tool divided into four levels of authoring: content creation, lesson definition, course management, and authoring environment. Content creation functions include the input and modification of text, graphics, sound or synthesized speech, and variables. Content creation represents the most time-consuming part of the authoring process, but it is not the most complex. Lesson definition is the most complex aspect of authoring of different type styles and graphics possible and the range of branching allowed.

However, relatively few authoring tools provide capabilities at the course management or authoring environment levels. This means that authors must take care of assembling lessons, specifying student control options or defining new strategies or templates themselves via an authoring general purpose language. In many cases authors are expected to become versatile with the particular operating system used (e.g., DOS, UNIX, CP/M) in order to perform these functions. As a consequence most authoring programs available today allow good quality instructional programs to be created within a limited range of traditional CBI strategies (such as drills, tutorials, simulations, etc.).

The limitations of authoring may be either accentuated or remedied by the design expertise and sophistication of the author and by several practical considerations in the authoring process.

**Practical Considerations**

Practical considerations are often much more serious limitations to the authoring process than instructional design constraints. Characteristics of the delivery system tend to dictate what features are possible. Table 2 lists some major system characteristics that determine the nature of a CBI program. Clearly, the availability of certain capabilities in an authoring program will depend upon the characteristics of the intended delivery system.

**Compatibility.** Authoring software is usually designed to be used with a specific computer. Pilot is a notable exception. This means that the instructional programs created will only run on the specific computer used for authoring. Furthermore, the pace of innovation in the computer field is frantic. Many developers have spent considerable time creating instructional programs for machines which were no longer popular or sold by the time they finished their programs. The dilemma is that good authoring software is usually not available for a machine until a year or more after its introduction in the marketplace.

**Usability.** It was noted in the preceding section that authoring software becomes more powerful in an instructional sense as additional functions are added. However, the price paid for the additional power is complexity. More powerful authoring languages or
Table 1:  
Authoring Software  
Characteristics

CONTENT CREATION

Text  
input mode (promoting, menus/fields, commands)  
formatting (margins, justifications, character size, fonts, word wrap, spacing, pagination)  
editing (line/page mode, globals, insert/replace/delete/copy)  
selective erase

Graphics  
simple line (boxes, circles, graphs)  
illustrations (drawing, digitization)  
dynamic (animation)  
character sets  
library

Sound/Speech Editor

Variables  
stored  
random

LESSON DEFINITION

Display Integration  
text, graphics, sound/speech  
multimedia (slides, video, audio)  
editing (move/delete/replace)

Student Response Processing  
response type (single character, M/C, string match, keywords, algebraic, touch/cursor position)  
answer logic (explicit/implicit, Ca/Wa/Un designation, feedback messages)

Branching  
unconditional (labels, function keys, restart)  
conditional (counters/buffers, arithmetic, logical)

Documentation (comments, objectives)

COURSE MANAGEMENT

Lesson Sequencing/Linking  
Lesson Try-Out Mode  
Response Data Specification  
Student Control Options (back/forward, helps, maps)  
Instructional Strategy Definition  
Documentation  
Message/Mailbox  
Debugging Aids

AUTHORING ENVIRONMENT

Template Definition  
Hardware Configuration Specification  
Language Translation  
Lesson Compression  
Uploading/Downloading (Networking)  
Helps/Training

Needed developments

The preceding remarks have alluded to a number of areas where advances in authoring software are needed. While the point has been made that currently available authoring software will allow good instruction to be created, software does not necessarily make instruction easy to create. For example, most authoring systems provide the capabilities needed to create simulations and games. However, the present collection of authoring languages and authoring systems do not facilitate their creation. As another example, consider test item generation. Most authoring software will help you create a wide variety of tests but not help you write test item generators.

A major deficiency of current authoring software lies in the domain of intelligent tutoring systems or ICAI (Sleeman & Brown, 1982). ICAI programs use artificial intelligence programming techniques to provide a "deeper" level of understanding about the content being taught and the student's learning progress. These programs involve knowledge networks which are acted upon by tutoring rules in order to generate student dialogues or problems. They are usually "mixed initiative" in nature in that either the student or computer can take the initiative at any time to ask a question or pursue an idea.

A number of ICAI programs have been constructed in the past decade as research paradigms. These include SCHOLAR, a geography tutor (Collins, 1977), SOPHIE, a tutor for electronics troubleshooting (Brown, Burton & deKleer, 1982), WEST, a coach for game playing (Brown, Burton, 1978) and STEAMER, an intelligent simulation (Stevens & Roberts, 1983).

As a consequence of these and other ICAI projects, it is now clear how to design more sophisticated CBI programs than the ones in current use. The problem is that the construction of these programs is fundamentally different from the design and development process we presently follow. Furthermore, the authoring process for ICAI programs relies heavily on the knowledge of artificial intelligence programming language (such as LISP or Prolog). If ICAI programs are to be developed and used more widely, authoring software is needed that can be used to develop such programs easily.

The biggest deficiency of current authoring software, however, lies in its...
narrowness. It was emphasized earlier in this article that the use of good design principles overshadows the importance of authoring software limitations. Most of the effort in developing instructional programs lies in the analysis, design, and evaluation stages. Therefore, if authoring software is to have an important impact on development, it should also assist in the analysis, design, and evaluation stages.

A number of research studies have been conducted in the area of automated instructional development systems. Braby and Kincaid (1981) describe a system for the design and production of technical manuals. O'Neal and O'Neal (1979) discuss a prototype authoring management system to be used in large instructional projects. Merrill and Wood (1984) describe their Lesson Design System for automating instructional design. At Courseware, Inc., a system called the Instructional Toolkit is being developed which automates the ISD process and runs on a personal computer.

This area of research should ultimately result in full-scale authoring systems which streamline the entire development process and improve the quality of instructional materials. Ideally, such systems should provide an integrated set of programs for the analysis, design, development, implementation, and evaluation of instruction of any media (not just CBI). To the extent that good instructional design principles are built into these programs, it should be possible for novice instructional designers or content experts to create high quality instructional materials. And, for instructional development systems should be a productivity tool, increasing their efficiency.

There is a rather profound connection between these two lines of research. Work on the Instructional Toolkit has shown that many aspects of the instructional design process can be automated by using the same programming methods underlying applications such as word processing, spreadsheets, and database management. However, many of the more complex aspects of design are very intuitive and heuristic in nature. In order to automate these processes, artificial intelligence programming techniques and theory will be needed. Thus, the design of authoring software for creating intelligent CAI tends to bring together these two research areas.

Conclusions

One of the intriguing aspects of attempts to automate instruction is that it forces us to examine the process of developing instruction to a much greater depth than in the past. Authoring software has forced us to think about exactly what features are desirable in a CAI program and the process of creating an instructional program. Intelligent CAI makes us look much more closely at what it means for a student to understand a concept or a subject. The development of authoring systems to create ICAI programs will require an even greater analysis of the process of creating effective instruction. Similarly, the development of automated instructional design systems requires a very detailed study of the instructional development process. Thus, the net effect of developing authoring software is that it makes us look deeper at our instructional design methodologies. Future developments in authoring tools will extend this analysis. For this reason authoring software is of both practical and theoretical significance.

The net effect of developing authoring software is that it makes us look deeper at our instructional design methodologies. Future developments in authoring tools will extend this analysis.

It is important to realize that advances in authoring software are primarily driven by new developments in computer technology. For example, the emergence of machines like the Apple Macintosh with high resolution graphics, extensive windowing, pull-down menus, and the mouse, mean new capabilities are now possible for CAI programs and, hence, authoring software. Similarly, the emergence of interactive videodisc and its continuing evolution mean new possibilities for CAI programs and authoring software. Because authoring software is so intimately associated with the capabilities of hardware and software, it is likely to always be in a state of change.

We are moving towards an era when computer literacy will be required of any instructional designer. This will mean the ability to use authoring software of one form or another to create instructional materials. It is likely that the better designers will be those individuals with the greater mastery of authoring tools. Our current authoring software is just a crude foreshadowing of things to come. Yet even these primitive programs can already help considerably in producing better quality computer instruction.

References


Burdesson, C. V. (1974). The design and production of learner controlled courseware for the

The only magazine devoted exclusively to instructional development, the *Journal of Instructional Development (JID)* is for those involved with the design, implementation and evaluation of courses or curriculum. JID contains articles on theories, techniques, reports, case studies, and critical reviews of instructional development projects and systems. It’s designed to stimulate communication among instructional developers at all levels of education and training.
Artificial Intelligence Methods in Computer-Based Instructional Design

The Minnesota Adaptive Instructional System

Robert Tennyson
School of Education
University of Minnesota
Minneapolis, MN 55455

Abstract. Design of computer-based instruction (CBI) is presented from a management systems perspective using methods of artificial intelligence (AI). Instructional design variables taken from a programmatic research effort based on the Minnesota Adaptive Instructional System (MAIS) are reviewed. AI concepts are an integral component of the MAIS. Instruction is iteratively adjusted for each learner according to at-the-moment learning needs. Following a brief introduction to AI methods appropriate for intelligent CBI, the six design variables of the MAIS are reviewed. A unique feature of the MAIS is the extensive collection of empirical research findings that support the application capabilities of the design variables.

For the past decade educational researchers have gradually moved from a behavioral learning theory base for the improvement of learning to a cognitive science base (Glaser, 1984; Newell, 1980; Scandura, 1984). This move has had a direct effect on the field of instructional development in terms of how we go about designing individualized systems of instruction. Recently, Gagne and Dick (1983) reviewed the research literature in instructional psychology and showed that current instructional theories have a definite cognitive science approach to learning. Concurrent with this transition period in learning and instructional theories, has been the movement in computer science research to the use of Artificial Intelligence (AI) methods in hardware and software designs (Feigenbaum, 1977). The assumption in AI designs is that the system will acquire not only an increase in the accumulation of knowledge but, with experience, improve decision making (Nilsson, 1971).

The purpose of this article is to review a programmatic educational research effort that has joined the methods of artificial intelligence to the development of a cognitive science-based instructional design system. This particular instructional system applies AI methods to the design of computer-based management systems that adapt the learning environment to individual differences and needs. The article will briefly review educational applications of AI, then present the empirically-based design variables for developing a management system for computer-based instruction (CBI).

AI Application

The formal study of AI in computer science can be traced to the early 1960s (Feigenbaum & Feldman, 1963). The focus was on the design of computer programs that would enhance decision making, as well as storing and retrieving information. Early attempts in cognitive science to simulate the brain with computer models piqued interest in how to simulate decision making by experts (Amarel, 1969). The application of AI in education came through cognitive science research on problem solving (Tennyson, 1982).

Although there are many forms of AI application (see Amarel, 1983), the two forms most widely used in education have been tree structure and heuristic models. Application of tree structures is readily seen in formal designs of "expert systems" in health sciences and industry. An expert system is basically an information retrieval system based on an expert analysis of a domain of information. The expert system structure of a given domain usually resembles a taxonomic form of closely networked concepts connected by subject matter attributes.

The other AI form which seems to lend itself directly to educational applications in CBI is the heuristic method (Lenat, 1982), built around direct connections to a cognitive science base of learning (Polya, 1945). A heuristic is a "rule of thumb" search strategy composed of variables that can be manipulated to provide increasingly better decisions as more knowledge is acquired. The method differs dramatically from the tree structure methodology of AI programming in that it is usually written as a conditional probability statement code. Also, a heuristic may be thought of as a higher order rule statement rather than a depository of domain-specific information or content.

Cognitive science theory suggests that higher order problem solving rules are more flexible than conditional rules which are useful under only specific and limited situations (Sternberg, 1981). In research dealing with CBI design variables at the University of Minnesota, we have focused on a heuristic approach for the instructional management system strategy variables because it allows for the growth of the system as we investigate simultaneously learning variables, instructional variables, and conditions. The heuristic approach makes it possible to increase the adaptability of the management system as we test additional design strategy variables that are linked to the improvement of learning. The next section of the article reviews those instructional design variables that allow for the development of adaptive CBI.

Minnesota Adaptive Instructional System

My first effort at designing an adaptive instructional model focused on the recognition of error patterns in concept learning (Tennyson, 1975). The error pattern recognition strategy attempted
to adjust the sequence of examples according to identified errors of overgeneralization, undergeneralization, and misconception (Woolley & Tennyson, 1971). This early attempt at response-sensitive sequencing of instruction has led to a broader description of a CBI management system that individualizes the instruction to learner differences rather than merely making it self-instructional. The basic structure of the Minnesota Adaptive Instructional System (MAIS) was proposed around Bayes' conditional probability method (Tennyson & Rothen, 1977). Bayesian statistics have been widely applied in testing (e.g., Novick & Lewis, 1977) and economics because the formula's parameters can be manipulated while using current data to predict future needs.

Other basic fundamentals of the MAIS include (a) iterative updating of the decision making parameters, (b) use of a variety of variables concurrently to form a diagnosis of learning needs (e.g., performance data on prerequisite knowledge, on-task learning progress data, individual differences data), (c) flexibility that would easily allow the addition, modification, and deletion of heuristics, and (d) transportability to any hardware and/or software system as well as subject matter. In summary, the goal of the MAIS research program was to develop an intelligent adaptive management system for CBI that would enhance decision making to improve learning. Additionally, my colleagues and I set about to test the variables of the system using rigorous, experimental methods. We sought to design an empirically based theory rather than a model-based system. A review of the specific research studies supporting our design variables is presented in Tennyson, Christensen, and Park (1984).

The following presentation of the Instructional design variables will be structured around two main sections. The first section concerns the Bayesian component of the MAIS and the learning theory and instructional theory upon which the MAIS is founded. The second section concerns the six main instructional design variables of the MAIS: amount of instruction, sequence of instruction, display time, advisement, refreshment of prerequisite information, and individual differences.

Theoretical Structure

The learning theory of the MAIS is philosophically grounded in cognitive psychology (Tennyson & Breuer, 1984).

The theory views learning of information in reference to acquisition of conceptual and procedural knowledge around a schema theory of memory storage and retrieval (Tennyson & Cocchiarella, in press). Acquisition of conceptual knowledge is primarily a function of exposure to information through expository experiences. Information encoded from the expository experiences provides initially sufficient conceptual knowledge for development of a schema to solve problems, thus developing procedural knowledge. As the learner experiences additional problem solving situations, the conceptual knowledge of the schema is further formed and the connections in the memory to other existing knowledge structures are further developed (Anderson, 1980; Scandura, 1977, 1984). As the knowledge base grows in the long-term memory around conditional problem solving, more creative higher order problem solving experiences become increasingly possible (Dorner & Rether, 1979).

Bayesian Conditional Probability

The Bayesian component of the MAIS provides the basic data structure on which several instructional decisions are made. Mathematically, Bayes' theory of conditional probability uses a set of parameters that allow increasingly better predictions as more data are acquired. Since the parameters of the statistic are continuously adjustable, it is possible to have almost infinite predictions according to individual learner differences. Complete reviews of the Bayesian theory are given in Rothen and Tennyson (1978) and in Tennyson and Rothen (1979).

The parameters of the Bayesian statistic include a criterion level, an error ratio, and a performance value. The calculation of these parameters produces a matrix of beta weights that provide the data source for decision making in several of the design variables. Values for each of the parameters is determined

The goal was to develop an intelligent adaptive management system for CBI that would enhance decision making to improve learning.

An important construct of the MAIS, although not directly part of the management strategy, is how the structure of the information to be learned is organized for instruction. Typically, content analysis methods follow a taxonomic approach such that the relationships of concepts in a content area are based on critical attributes. A taxonomic structure of content seems to be the way individuals store information and the way they retrieve it when asked to recall relationships between concepts (Rosch, 1978). However, the taxonomic structure of information fails to provide the conditions necessary to use the knowledge to solve problems (Mandler, 1979). A taxonomic presentation seems to allow for the foundation of conceptual knowledge but not procedural knowledge. An alternative approach to the taxonomic method of content analysis is a schematic analysis (van der Waerden, 1973). According to both individual learner differences and program considerations.

The criterion level establishes the rate of performance desired at the conclusion of the instruction. It is not the same as a criterion level usually associated with testing (e.g., on a behavioral objective). Testing at the conclusion of instruction assumes that learning has occurred and that the measurement is a true assessment of what has been learned. A criterion level in instruction must assume and, therefore, account for errors in learning during the entire learning process. Bayes' theory is excellent in this regard because it weights early errors progressively less and less as the learner advances in learning. Unlike a percentage statistic between correct and incorrect solutions, the Bayesian procedure is able to increase its power of predictability rapidly as the learner acquires sufficient procedural knowledge to solve problems correctly.
The error ratio parameter, technically termed a loss ratio, determines the balance between falsely advancing a learner who has not actually learned to criterion, as contrasted to falsely retaining a learner who has in fact reached the criterion. This parameter value can be adjusted to the specific learning situation and/or individual differences. A higher error ratio value would require more evidence of correct solutions than a lower value. Using the heuristic method, the error ratio value can be adjusted as the MAIS accumulates experiences.

The third parameter, performance value, assesses performance by comparing the number of correct solutions to the number of problems presented. We have found that by varying the number of problems through simulations of data runs, that a maximum number of problems for learning concepts is 14. We have also found that a minimum of four solutions per concept or rule is necessary before predicting mastery at any given criterion level.

**Instructional Theory**

The main instructional focus of the MAIS is to increase the amount of interrogatory learning experiences in reference to expository experiences. Most formal instruction uses a learning experiences ratio of 70 percent expository to 30 percent interrogatory. In the MAIS, we attempt to reverse that ratio (L'Allier, 1984). Program designs usually follow branching methodologies that set arbitrary and static sequences of objectives, that allow for only a finite set of decisions to account for learning problems. The goal of the MAIS research program, in direct contrast to the conventional approach to CBI design, was to define a set of design strategies that would (1) allow for continuous adapting of instruction to individual learner needs, (2) provide an almost infinite means of presenting information, and (3) would respond intelligently to learner needs progressively better during instruction.

The MAIS is currently composed of six design variables that focus on the improvement of learning through intelligent management of the instruction. The six design variables are: amount of instruction, sequence of instruction, display time, advisement of learning progress, refreshment of prerequisite knowledge, and adjustments of instruction to individual differences. MAIS does not deal directly with specific display characteristics, such as use of color, graphics, and display layout. These variables are important elements of instructional design and we use them in the design of CBI.

**Amount of Instruction**

Determining the amount of information to provide a learner is a primary function of an instructional system. Because of differences in background knowledge, prerequisite knowledge, prior knowledge, and aptitude, each learner requires a different amount of instruction to learn a given domain of information. For example, in our research findings (Tennyson & Rothen, 1977), we have shown that if learners receive more instruction on a concept than is necessary, performance actually deteriorates. And, of course, if insufficient information is provided, then the level of learning is limited.

Basically, a minimum presentation of expository information is provided initially. Our learning theory states that learners will attempt to solve problems first with existing knowledge, and when faced with the awareness that additional knowledge is necessary, proceed to acquire additional information. Thus, the amount of instruction is determined when the learner attempts to problem solve. A minimum amount of expository information is used to establish a working schema that problem solving with interrogatory instruction can elaborate.

It is during interrogatory instruction that a given schema is learned and connections with existing knowledge occur. The expository instruction that precedes interrogation consists of a statement of the problem area or context, the label and definition of relevant concepts or rules, and a best example, all of which initializes the schema. Exposure to problem solving experiences begins immediately after the expository instruction. Once in the interrogatory section of instruction, the amount of instruction is determined by the three parameters of the Bayesian method.

By adapting the amount of instruction to individual learning progress, an increasingly intelligent decision is made rather than setting an arbitrary amount of instruction. And, because the heuristic nature of the decision for amount of instruction includes adjustments to individual differences as well as context conditions, these adjustments can be increasingly refined from experience data. Closely associated with amount of instruction is the sequencing of the information.

**Sequence of Instruction**

Early attempts to provide response-sensitive sequencing of instruction followed decision algorithms based on mathematical or statistical probabilities (e.g., Atkinson, 1976; Hansen, Rakow & Ross, 1976). These procedures were limited because they are based on a system artificially independent from learning theory (Tennyson & Park, 1984). In contrast, we designed and tested a procedure directly connected to a theory of learning (Park & Tennyson, 1980). The response-sensitive sequence design variable of the MAIS uses a heuristic that adjusts the flow of information continuously according to learning needs at each given moment.

Sequence decisions are made during the interrogatory section of the CBI lesson based on assessment of each learner's response to a problem. For correct solutions, it is assumed that the learner understands the concept or rule,
and needs additional problem solving to learn procedures. Additional problems are presented until mastery is reached.

If a learner either fails to provide an answer or provides an incorrect solution, a response-sensitive decision is made. In either case, it is assumed during initial learning that learners need to focus on conceptual understanding; therefore, the instruction is narrowed to only one concept or rule, even though a given lesson may have several coordinate concepts or rules. In this case, the next problem would be presented from the same concept or rule class used initially. Conceptual understanding implies the ability to generalize within a given domain of information; therefore, the instructional sequence strategy focuses the learner on the conditions of problem solving within a given concept or rule.

Assessment of mastery occurs after a minimum number of responses are made; this initial period ranges from 4 to 6 problem solutions. At this point, the amount of instruction is determined iteratively, and the sequence rule focuses on discrimination decision making as well as correct solution behavior. For incorrect solutions, the sequence decision computer was a patient tutor has led many developers to assume that worrying about time is not necessary. This represents a continuing use of programmed instruction (PI) approaches to CBI design as well as a misunderstanding of the mastery learning concept of sufficient learning time.

From a practical point, no learning situation is infinite in time. All learners have real constraints on time allocations. Quality instruction makes it possible to have efficient learning environments so that as much as possible can be learned within the time available for instruction. This is especially true for slow learners.

Likewise, understanding the learning process more fully indicates an instructional need to attend more directly to the monitoring of learning time. First, empirical findings on learning show that the initial period of learning is the most active time for acquisition of information. That initial period is approximately 20 minutes for secondary school age students. For younger learners the average time is probably shorter while for adults a longer period may be expected. Regardless of the length of that initial period, the learning environment

To allow this design variable to exhibit intelligence, we have further adjusted the information presented and options to the learner when time expires (Tennyson & Park, in press). The presentation of other information is discussed below in the refreshment of existing knowledge section. Additional options include student control over the decision to resist the timer and adjusting the normed times to individual differences (Tennyson, Park, & Christensen, in press).

Advisement

An early recognized failure of learner-controlled CBI systems was the finding that learners were not good at decision making in reference to how much and what kind of instruction they needed. To effectively deal with these two problems, we introduced the concept of advising the learners continuously of their needs and progress in learning (Tennyson, 1981). Conventional CBI lessons, unlike a workbook, are not good in visually helping the student see the entire amount of instruction prior to and/or during instruction. The learner in most CBI situations is without a means for judging where they are in acquiring the information from a specific CBI lesson. For example, with a workbook, the learner can see how many pages need to be done; they can see lines being filled in or problems being answered. They can actually see their progress in finishing the work. One assumption in a schematic organization of information is that prerequisite knowledge can be readily recalled so that appropriate connections between existing knowledge and the information to be learned can be made. Typically, refreshing prerequisite knowledge is done in a review prior to instruction—if it is done at all.

Prior refreshment of prerequisite knowledge seems to contribute minimally to learning from a cognitive science view. First, the information recalled in a review is not kept in working memory once instruction begins. And, secondly, the presentation reviews the prerequisite knowledge without benefit of the connections to the new information.

Recently, we have investigated a means for refreshing specific prerequisite knowledge at that point in the instruction where the learner needs help in making connections with new knowledge (Tennyson, Welsh, Christensen, & Hajovy, in press). This instructional

A constant criticism of computer-based instruction is the failure to provide real-time individualized instruction.

is to present the next problem example from the same concept or rule class the learner incorrectly solved. The purpose is to focus on the decision process of selecting the appropriate concept or rule, not simply on producing a correct solution. Theoretically, the assumption implies conceptual understanding, with the need being to learn procedures.

Display Time Interval

One of the most overlooked design variables in CBI development is the means for controlling the allocation of learning time. Unfortunately, many developers of CBI materials have misconceptions about learning time. Most approaches to CBI design are direct violations of good pedagogy (Walberg & Tsai, 1984). Many CBI developers assume that learners have unlimited time to learn and that learners can make good use of their time in learning new information. The idea that the needs to be immediately active. The transition from expository to interrogatory instruction will not be the same for each learner. Learners with insufficient conceptual understanding of a concept or rule will be unable to understand the problem situation or to propose a solution. We investigated a design variable that permitted individual transition between expository and interrogatory instruction by sensing a lack of conceptual understanding (Tennyson & Park, 1984). This was done by monitoring the display time interval so that if the program sensed a no-response situation, it would interrupt. In our first test of the display time interval, each interrogatory example was normal and when time expired without a response, the correct solution was provided. This allowed the learner an opportunity to receive more conceptual knowledge without the fear of forcing responses and possibly encoding incorrect knowledge.

20 JOURNAL OF INSTRUCTIONAL DEVELOPMENT
design variable, embedded refreshment, presents the specific prerequisite information only if the learner is unable to solve an interrogatory problem. Operationally, the embedded refreshment helps the learner recall the prerequisite knowledge, retrieving it for use in working memory simultaneously with the acquisition of the new information. Embedded refreshment offers help in both recalling prerequisite knowledge by placing it in working memory and by making the connections between the existing knowledge and the new knowledge to be learned.

Implied in embedded refreshment is the assumption that prerequisite knowledge is in long-term memory and that recalling it helps in learning new information. It is not a means by which the learner can at that point learn the prerequisite information. We have two associated design variables to assess and assist the learner in reestablishing a mastery level use of the prerequisite knowledge.

The first variable is a pretest that evaluates the learner's current mastery level of the prerequisite information. Thus, a learner would not even begin instruction until mastery was assessed. This system of testing actually presents instruction if the learner is assessed not to have the defined mastery level. The second variable is a procedure where the MAIS senses that the learner needs more than just the embedded refreshment of specific prerequisite knowledge. At that point, the learner is provided with remedial refreshment that temporarily removes the learner from the main program. The remedial refreshment provides additional instruction in the form of interrogatory practice. Again, we assume that refreshment is used to retrieve from long-term memory appropriate prerequisite knowledge for use in working memory. Remedial refreshment is provided for only specific situations and not for initial encoding.

Individual Differences

One of the first hopes of computer technology applications in education was the individualization of instruction according to individual learner differences. However, this hope is yet to come about because of two factors. The first is that attempts to identify the interactive effects of individual difference variables with instructional variables has been elusive. Too often experimental designs have approached the interactive effect with one possible individual factor crossed with one instructional design variable. For example, testing a learning style variable (e.g., field independent vs. field dependent) with a concept teaching strategy (e.g., best example vs. definition). When significant findings are reported, they usually appear to be situation specific and not generalizable.

In our research, we view the study of individual differences from a multiple regression approach so that the contribution of a number of individual differences variables to interactions can be observed.

A second factor is the microcomputer and its limited memory capacity which moved instructional design away from management systems that could account for individual differences. Before microcomputers, there was much work on computer-managed instruction (CMI) (Tennyson & Park, 1984); but mostly for data storage and retrieval. The microcomputer revolution interrupted the serious study of using large amounts of data for instructional decision making.

For CBI, the value of individual differences can be appreciated from a curricular level. For CBI to be intelligent, it needs to accumulate information over time. For the next several years, our goal is to investigate management of CBI over time at the curricular level. Individual differences variables that we feel need testing range from personality to cognitive to biopsychological. Basic research in each of these areas indicates a high potential for application (Farley, in press).

Summary

The purpose of this article was to identify instructional design variables that contribute to the development of intelligent CBI. The variables reviewed were part of a program of research that has brought together theories and research from the fields of learning, instruction, and educational technology. This interdisciplinary approach has allowed for the development of an adaptive instructional system that exhibits intelligence in instructional decision making.

An integral philosophy of the MAIS is that intelligence assumes a continuing advancement in decision making as knowledge is acquired. To implement this philosophy, the computer science method of heuristic programming has been used. Even as new variables are investigated we can continuously improve earlier variables with experience. At the present time, the MAIS operates at the instructional level of management: adapting instruction only after the learner enters the program and the program starts acquiring information. Future research will focus on the curricular level so that adaptation can begin before instruction begins and continuously interact between the curriculum and instruction levels.

References

Peigenbaum, E. A. (1997). The art of artificial intelligence. *Proceedings of the 5th International Joint Conference on Artificial Intelligence,* Cam-

One of the most overlooked design variables in CBI development is the means for controlling the allocation of learning time.

1984, VOL. 7, NO. 3
based adaptive instructional strategies. Educational Psychology, 12, 317-332.
Alternative Designs for Evaluating Computer-Based Instruction

F J King
Professor, Florida State University
and
M. D. Roblyer
Associate Professor, Florida A & M University

Abstract. As more computer-based materials and methods appear in the educational marketplace, there is an increasing need for studies to validate the effectiveness of these products of technology and to provide more substantial guidelines for future development. However, efforts to study the usefulness of computer-based methods and materials with students are rare in comparison with the opportunities for such studies. Although many reasons have been hypothesized for why more research on the effects of various computer-based methods and materials has not been accomplished, two which seem most likely are: (1) practical difficulties in arranging studies with traditional randomized two-group designs in non-laboratory settings, and (2) lack of available expertise to develop and employ alternative designs and analyze resulting data. This paper:

- Presents five designs (three one-group and two two-group) which can be effectively employed when a randomized two-group design is not practical
- Gives examples of evaluation studies which effectively employ each design
- Provides references to procedures which can be used by persons with limited training in statistics and evaluation methods
- Gives a decision flowchart and describe how to select the most appropriate alternative design for a given project

Design #1: Sequential Analysis
Sequential analysis originated almost forty years ago (Wald, 1947), but, with few exceptions, it has been used for quality control of manufactured products rather than to evaluate the quality of educational materials. It can be used to construct an effective design for evaluating instructional materials. This technique could be especially useful to school administrators responsible for developing or purchasing computer-based programs for district-wide distribution and use. Before they make the considerable expenditure of time, money, and resources required to implement computer-based programs, administrators should have some data to indicate their instructional effectiveness. They may find it logistically difficult to arrange for a control group and to have an appropriate number of students, copies of materials, and computers for such a study.

An administrator would find that sequential analysis has many advantages for evaluation of computer-based materials besides not requiring a control group. Its principal advantage is efficiency. Decisions can be made with as much as 50% fewer observations than would be required for procedures in which the sample size must be specified in advance. This makes it an ideal choice
when the number of people who work with the materials is limited by factors such as the number of computers available, by not being able to obtain a large student group at one time, or by the expense of implementing the program.

In such studies, the following strategy would be employed: (a) an observation would be made (e.g., one or more students are tested on some skills), (b) the results are recorded on a chart or graph developed for this purpose, and (c) by reading the chart, a decision is made to reject the instruction, to accept the instruction, or to make another observation.

There are some limitations on when sequential analysis can be effectively used. Since there is no control in this design for prior learning, it is better used when there is little likelihood that students would be able to perform the skills involved before exposure to the materials. It is also desirable that the skill units be relatively short, to reduce effects due to maturation and history (Campbell & Stanley, 1963). If the materials under study are lengthy, they can be divided into short units, and a sequential analysis done on each one. This technique is also most useful when teachers want to make absolute judgments about materials ("Are these computer-based units effective?"") rather than a comparative one ("Are these lessons better than other non-computer lessons?").

Procedures to Implement Sequential Analysis

The following procedures are required. For detailed explanations of the procedures, see Epstein (1975), Epstein and Knerr (1978), Wald (1947), and Burr (1980). Lathrop (1983) also gives a detailed description of statistical procedures required for sequential analysis, although he used it for evaluating student performance rather than instructional materials.

**STEP 1:** Set acceptable/unacceptable quality level. Instructional effectiveness is defined in terms of the preparation of students who successfully complete the instruction. Since absolute accuracy in determining this proportion cannot be achieved with a sampling technique, it must be translated into a range of proportions which define acceptable and unacceptable instructional quality.

**STEP 2:** Set risk parameters. The probability of rejecting high quality instruction (Alpha) and the probability of accepting low quality instruction (Beta) must be set.

**STEP 3:** Construct Operating Characteristics Curve (OC), Average Sample Size Curve (ASN), and truncation number. An OC curve is constructed which will show the probability of rejecting or accepting the program for any true value of the success proportion under consideration (given the specifications of the test). Then the ASN, or the average number of student scores needed to reach the decision for any true value of the success proportion, is calculated. Finally, the truncation number is calculated, which is the largest number of students that could be needed to make a decision without altering the risks the experimenter is willing to accept.

**STEP 4:** Construct graph for recording observations. The graph for recording observations contains three zones corresponding to the three possible decisions which are considered each time an observation is made (to accept, reject, or take more samples). The actual decision choice is determined by the pattern of recorded data. Students are tested, one by one, until the decision is made to accept the instruction, reject the instruction, or until the truncation number is reached. If the truncation number is reached, the instruction would be accepted if the last point was nearer the line indicating acceptable instruction, or it would be rejected if the opposite prevailed.

Example of Sequential Analysis Use

A study using the sequential analysis method is currently being undertaken by the U.S. Army to validate the effectiveness of microcomputer/videodisc training materials (King, n.d.). The modules were designed to teach specialist students to perform a lengthy and expensive procedure for testing the purity of jet fuel. No other training is currently being given, so no control condition is available. A sampling procedure such as sequential analysis is an appropriate design for this evaluation situation since the length of the instruction and the limited number of videodisc units available would make it difficult to test a large group of students at once. Effectiveness is to be defined in terms of the proportion of students who successfully complete the simulated task in no more than two trials. In this case, the proportion is set at .80.

Design #2: Value-Added Analysis

Many computer-based instructional materials are large-scale programs which deal with such areas as reading and writing skills and motor skills development. A great deal has also been written about the impact of LOGO on problem-solving skills. Although the expense of implementing these programs can be considerable, and, in the case of LOGO-related activities, may call for revolutionizing classroom methods, few studies have been done to measure the impact of these materials. School personnel wishing to gather evidence of effectiveness before investing in such innovation will find that study of these materials must often extend over a long period of time in order for measurable effects to occur. The value-added analysis is appropriate for evaluation of these long-term activities, since it controls for natural maturation processes.

The basic idea underlying value-added analysis is that the effect of instruction can be estimated by comparing the average observed growth between pretest and posttest with the estimated growth that is expected to occur in the absence of instruction (Bryk & Weisberg, 1976). An unbiased estimate of growth rate can be provided by way of ordinary least square regression of pretest on the student's age, if age at the time of pretest is not related to systematic growth. When growth (estimated from a cross-section of the population) is not linear, transformations must be made before value-added analysis can be completed.

Certain conditions must be present in order to use value-added analysis. An independent variable which can be measured without error (i.e., chronological age) must be available, and it must be correlated with pretest scores. A large enough number of students must be available to be able to compute a product-moment correlation. In addition, it must be assumed that longitudinal growth can be estimated through study of a cross-sectional sample.

Procedures to Implement Value-Added Analysis

Value-added analysis involves the following steps. For formulas and detailed descriptions of statistical procedures, see Bryk & Weisberg (1976) and Bryk, Strenio, and Weisberg (1980).

**STEP 1:** Estimate of growth rate. First, an unbiased estimate of growth is obtained. This is done by regressing
pretest scores on age and using the resulting regression coefficient as the estimate of growth rate. This figure is then entered into a formula to obtain an unbiased estimate of growth rate.

**Step 2: Calculate value added.** The value added by the instruction is calculated using a formula to determine the effect of the instruction which takes into account the natural growth of students over time. To increase the precision of this model, Blyth et al. (1980) also describe how it can be extended to incorporate background variables that may be related to individual growth rates. This is done by regressing the pretest on age and the first order interactions of age and the background variable.

**Example of Value-added Analysis Use**

This procedure was used in a study of a microcomputer-based reading and writing program in grades K-1 (Carretson, Vertuno, King, & Roblyer, 1983). All students in these grades (approximately 100) received the computer-based instruction, which was to take place over the school year. Several measures were used to assess skill levels, including a Nonsense Words Test devised by the school resource teacher and several subtests of the California Test of Basic Skills (CTBS). The value-added analysis was used with the Nonsense Words Test. (Analysis methods for the other test will be discussed later under another design.) The results indicate that the instruction resulted in significantly more learning than that expected from normal maturation.

**Design #3: Non-Equivalent Dependent Variables (NEDV)**

This design, like sequential analysis, is a time-saving evaluation procedure. Very often, schools want to evaluate the effectiveness of not one but several computer-based programs by a vendor. The NEDV design would allow evaluation of several programs at once. For example, suppose school personnel wanted to evaluate a computer-based module on fractions as well as one on geometry. Students could be given pretests on both units, instruction on fractions only, and then posttests on both units. If statistically and educationally significant changes occur on the fractions unit but not only the geometry test, the fractions unit is judged to be effective. Then instruction on the geometry would be given, followed by a test on geometry. If the change from the first to the second geometry tests is not significant, but change from the second to the third is significant, then the geometry unit is judged effective.

Of course, the evaluators would have to be satisfied that the tests were equally reliable, and that students were not already scoring at the ceiling of either test at the time they were pretested. In addition, the investigators would have to be satisfied that the change in performance in either case was not due to some learning experience other than the modules, such as some parent tutoring on this topic during the instructional period.

Such an approach has two advantages. First, no non-instruction control group is needed, since performance on one variable acts as a control for the other. Second, older students can be used, since, unlike the value-added analysis, no independent variable such as age is required. This design may be diagrammed as:

\[
\begin{align*}
&\text{O}_{1A} & \text{X}_A & \text{O}_{2A} & \text{etc.} \\
&\text{O}_{1B} & \text{O}_{2B} & \text{X}_B & \text{O}_{3B}
\end{align*}
\]

where \( \text{O}_{1A} \) and \( \text{O}_{2A} \) are pre- and posttests for the variable that is hypothesized to be affected by the instruction on fractions, and \( \text{O}_{1B} \) and \( \text{O}_{2B} \) are pre- and posttests for the variable hypothesized not to be affected during the first treatment period.

In order to be sure this design will be credible for the situation, the evaluator must (a) specify in advance which variables are expected to change and which are not; (b) demonstrate that differential change is not due to differential reliability of measures or to ceiling/basement effects; and (c) use variables which are conceptually similar that both would be equally affected by the same threats to internal validity, namely maturation and testing.

Berquist and Graham (1980) expanded this design to allow the evaluation of many objectives over many waves of measurement. They point out that this approach is useful to evaluate instruction with single students as well as groups. For details about how this approach is most appropriate, see Berquist and Graham (1980).

If more than one unit of instruction is to be evaluated and the group is sufficiently large (e.g., 60 students), this design can be converted to a true experimental design which does not require a non-CAI control group. Where instruction can be individually administered to each student (as with CAI), the following design could be used:

\[
\begin{align*}
&R & 0_{1A} & 0_{1B} & X_A & O_{2A} & 0_{2B} \\
&R & 0_{1A} & 0_{1B} & X_B & O_{2A} & 0_{2B}
\end{align*}
\]

Here, students must be randomly assigned (R) to either fractions or geometry (\( X_A \) or \( X_B \)) and pre- and posttested with both unit tests (\( O_{2A} \) and \( O_{2B} \)). Then two separate analyses must be done: one in which the group that received the geometry instruction serves as the control group for the fractions instruction and the other in which the fractions group serves as a control for the geometry instruction.

This design is really better in some ways than one having a non-CAI control group because it is a better control for Hawthorn effects. The novelty of the CAI experience is the same for both groups so that the computer aspect alone cannot be expected to produce significant achievement differences between groups.

**Procedures for Non-Equivalent Dependent Variables Design**

See Cook and Campbell (1979) for a discussion of the analysis of this design. Briefly, steps to implement the NEDV design are as follows:

**Step 1: Randomly assign and pretest students.** Students are randomly assigned to two instructional groups, and both are pretested on both modules.

**Step 2: Give instruction.** One group receives instruction in fractions, and the other geometry instruction.

**Step 3: Compare posttest means.** Students are posttested and the means of the two groups are compared to evaluate the effectiveness of the fractions instruction. The same analysis is done to evaluate geometry instruction.

**Example of True Experimental Design Involving Non-Equivalent Dependent Measures**

One study using this design as part of its total evaluation program was the ETS/LAUSD study of computer-assisted instruction and compensatory education (Ragosta, Holland & Jamison, 1982). Schools in the Los Angeles Area School District used math and language CAI available from the Computer Curriculum Corporation. In Grade 4 students were randomly assigned to
receive two sessions of CAI daily. They received either: (a) two sessions of mathematics (MM), (b) one session of reading and one of language arts (RL), or (c) one session of mathematics with one session of reading or language arts on alternate days (MRL). All students were pretested in the fall with the Iowa Test of Basic Skills (ITBS) and a curriculum-specific test (CST) in mathematics, language, and reading. They were posttested the next spring with the California Test of Basic Skills (CTBS) and the CST. Thus, the RL students were controls for evaluating the effectiveness of two levels of mathematics instruction (MM and MRL), while the MM students were controls for the evaluation of two levels of reading (RL and MRL). A regression analysis was used to analyze the data.

Although this was a complex study with many kinds of groups and taking place over several years, the results were generally positive for both math and reading groups. Math results were usually higher than for reading.

Design #4: Regression-Discontinuity

Although this is the one design which actually requires a no-instruction control group, the regression-discontinuity design is especially useful when compensatory or enrichment programs are to be evaluated. This is because of its requirement that students be pretested and all students below (or above) a certain score level be placed in an instructional group. For example, suppose a school district purchased a computer-based remedial math program with Chapter III compensatory education funds. Any students not in the compensatory Chapter III program could serve in the control group. Before any instruction is given, all students are pretested. Those whose scores fall below the cut-off level are placed in the instruction group. Those above this level are placed in a non-instruction group. At the end of the instruction time, the posttest is administered to all those who received the pretest. If the difference between the adjusted means of the groups is statistically significant, the instruction is effective.

The principal advantage of the regression-discontinuity design over other quasi-experimental designs is that it is not biased by errors of measurement in the pretest. That is, in the absence of an instructional effect, the expected posttest adjusted mean difference is zero. That is not true in designs in which unknown selection variables operate in the formation of instruction and control groups.

A primary disadvantage of the design when compared to a randomized instruction-control group design is that it has low power to reject the hypothesis of no differences between groups. If a randomized design used a sample of 100 students, the regression discontinuity design would require 275 students to have equal power (Reichardt, 1979). In addition, if one of the groups is relatively small, the estimate of the regression of posttest on pretest for that group may be unstable.

Two other precautions should be kept in mind when using this design. First, the process of assigning students to groups should be done without error. If the pretest is to be used as the covariate in the analysis, selection should be based on it only. If other sources of information, such as teacher or parent judgments, are used in addition to or instead of the pretest, the analysis may be seriously biased. For a further discussion of these problems and some ways of remedying them, see Huisema (1980) and Reichardt (1979).

The second precaution in using this design concerns the fact that non-linear relationships between pre- and posttest, which result from selection-motivation factors, may cause spurious treatment effects to occur. Reichardt (1979) suggested that the first step in analyzing data from this design should be to plot the raw data to see what relationships are suggested. If non-linear or interactive effects are suggested, the analytic model given previously can be expanded in an attempt to account for them.

Procedures for implementing this design are detailed in Huisema (1980). They are outlined as follows:

**Step 1:** Administer the pretest and group students.

**Step 2:** Give instruction and posttest students.

**Step 3:** Assign treatment variables. A "dummy" treatment variable is constructed by assigning a score of "1" to each member of the treatment group and a "0" to each member of the non-treatment group.

**Step 4:** Perform regression analysis. An analysis is performed using the following model:

\[ Y = a + b_1 T + b_2 \text{Pre} \]

where \( Y \) is the adjusted posttest score, \( T \) and \( \text{Pre} \) are the treatment and pretest variables, \( a \) is the regression intercept, \( b_1 \) is the difference between the adjusted means of the treatment groups, and \( b_2 \) is the regression coefficient for the pretest. If \( b_2 \) is statistically significant, the instruction is considered to be effective.

Example of Regression-Discontinuity Use

No example of this design in evaluation of instructional materials could be found in the literature. However, Cook and Campbell (1979) cite a study by Seaver and Quarton (1973) which examined how students' grades in one quarter were affected by making the Dean's list based on grades from the previous quarter. The investigators obtained the grades of a group of students during both quarters, and placed students in groups according to who did and did not make the Dean's list. Students who made the list were expected to do better than they would have done if they had not had the recognition of being on the Dean's list. Cook and Campbell (1979) interpreted the results of the study as not supporting the hypothesis because of the curvilinear trend in the data.

Design #5: Cohort Design

Cohorts are groups of persons who follow each other through an institution. For example, last year's first graders are cohorts of this year's first graders. The cohort design compares the performance of students in one instruction group with their cohorts who did not receive the instruction. This design is especially useful in evaluating the effectiveness of newly-installed, year-long computer-based programs where it is not feasible to withhold the program from some students in order to have a control group. Suppose a school district wished to study a popular and highly publicized new computer-based reading program for a given grade level before they decide whether or not to continue it in subsequent years. Attempting to select one or more classes which will not get to use the package the first year may be a political problem when parents and teachers find out what has happened. With a cohort design, all students in the grade level can use the package, if desired.

The design makes the assumption that student cohorts are comparable in most respects, and that differences between them on a dependent variable can be attributed to instruction administered to one of the groups.
In its simplest form, the design is as follows:

\[ 0_1 \times 0_2 \]

\( 0_x \) represents, for example, a final achievement measure of students in a certain grade in a given year, and \( 0_{y} \) represents the same final achievement measure of students in the same grade the year after that, after having received some experimental instruction. If a t-test of the differences between means of \( 0_x \) and \( 0_{y} \) yields a statistically significant result, and that difference is large enough to be considered educationally significant, the instruction is said to be successful.

A disadvantage to this design is that its use depends on the yearly availability of appropriate equivalent measures. Also, Cook and Campbell (1979) indicated that the design in this form is weak because differences between the two groups (other than the instruction) could cause a difference in achievement (selection). Events other than the instruction could depress achievement for the control cohort or enhance it for the instructional cohort (history). Variations in testing conditions or procedures could also be responsible for differences in performance. The design can be strengthened by adding a pretest if it is available. This augmented design is shown as follows:

\[ 0_1 \times 0_2 \times 0_3 \times 0_4 \]

The pretest could be an end-of-year test or one that is routinely administered in the fall. The data would be analyzed using regression analysis, the same method used in the regression-discontinuity design. The cohort design is considered to be superior to some designs in which the control group is not equivalent to the experimental group (e.g., the groups are obtained from different schools). The populations from which the cohort were drawn were probably more similar than those using students from different schools. Also, the school environment (principal, teachers, facilities, etc.) for the cohorts would probably be more similar than would the environments in different schools.

Procedures in Using the Cohort Design

**STEP 1: Calculate means and standard deviations.** The means and standard deviations of both groups on both measures are determined.

**STEP 2: Perform regression analysis.** This analysis actually involves several steps. For a detailed discussion, see Cook and Campbell (1979) and Huitema (1980).

Example of Cohort Design Use

Garrettson, Vertuno, Roblyer and King (1983) used this design in a study of a computer-based reading and writing program, the "Writing to Read" program from IBM, Inc. There were approximately fifty students in each grade (K-1), and it was not considered feasible to withhold the instruction from any of them. A readiness test was available for both the instruction and control kindergarten cohort, so it was used as a pretest. No pretest measure was available for the first grade students. The dependent variable was the reading score (prereading for kindergarten) from the California Test of Basic Skills (CTBS) administered at the end of both kindergarten and first grade.

The results for neither grade showed a significant advantage over instruction from the year before. These findings are not in agreement with those in the value-added analysis performed in the same study in which the instruction effect for the Nonsense Words Test was significant.

**Discussion and Summary**

Depending upon the needs of the evaluator and the characteristics of the situation, each of the five designs discussed here can serve as a practical alternative to a design requiring a no-treatment control group. Several caveats are, however, in order. Selecting the best design for the need and using appropriate procedures during the study are of critical importance.

Choosing the Appropriate Design
Each of the designs described here can be appropriate for studying various aspects of computer-aided instruction. But like all evaluation designs, each has assumptions which must be met and requirements which must be considered if the research is to be perceived as valid. The flowchart shown in Figure 1 can be used to choose the most appropriate design for the situation.

After selecting the most useful of these designs, the researcher must recognize that, like all designs, each has its strengths and weaknesses. Using two of these designs together can work to make the total study stronger. For example, in Garretson et al. (1983), both the cohort design and the value-added analysis were used. Ideally, the results of these two designs would be the same and thus support the same conclusions.

**Specifying Power**

The sequential analysis method is the only one of the five which forces the evaluator to set Alpha, Beta and effect size prior to conducting an evaluation. However, this practice is desirable for all designs (including true experiments) in order to determine that the study has sufficient power to reject the null hypothesis when it is false or to allow acceptance when it is true (Cohen, 1969).

**Required Expertise**

It should be apparent that using these designs requires some expertise in both evaluation methods and statistics. The minimum training would probably include introductory courses in research methods, descriptive statistics, and inferential statistics. Some experience in implementing evaluation studies is also helpful. Those with such expertise available to them, the designs described here can and should be useful in evaluating the effectiveness of computer-based instruction and should help direct the course of computer use throughout the field.

**References**


Crandall, N. (1977). An analysis of the impact of CAI on a program designed to ameliorate the ef...
Figure 1. Decision flowchart for choosing an appropriate design.


King, F. J. (n.d.) A plan to evaluate a microcomputer videodisc system to teach the jet fuel thermal oxidation test (JFTOT). Final report submitted to the Battelle Columbus Laboratories.


1984, VOL. 7, NO. 3
ERIC Reports on ID

Barbara B. Minor
ERIC Clearinghouse on Information Resources
Syracuse University


Concerned with curriculum theory and development and the place of curriculum theory within the area of instructional technology, this paper first identifies the central questions of curriculum (e.g., what should we teach?) and discusses the implications of the resulting issues: (1) Why should we teach this rather than that? (2) Who should have access to what knowledge? (3) What affects would accrue from the study, particularly the prolonged study, of a given domain of knowledge? and (4) How should the various parts of the curriculum be interrelated in order to create a coherent whole? The question of curriculum is examined and implications for the field of technology that might enhance the utilization of media within the instructional process are viewed. Conclusions indicate that linking the notions of curriculum and media together will suggest new ways of looking at the learning process and provide a different language and conceptual framework for looking at the issues, problems, and concerns in the field. Ten references are listed. —Microfiche 97 cents, paper copy $2.15 plus shipping, as document ED 243 427.


Surveys and follow-up interviews were conducted in 1978 and 1980 respectively at the Navy's Instructional Program Development Centers to determine the need for the development or modification of authoring aids (manuals containing detailed procedural guidelines) to support designers/developers in producing high quality, usable, instructional materials. In addition, tri-service availability and utilization of authoring aids were assessed. Results indicated that instructional strategy selection, terminal/terminal objective writing, and test instruction needed support. The tri-service assessment showed that existing aids and those under development would require major modifications to meet Navy requirements. It was recommended that: (a) designers/developers be encouraged to take courses/ workshops in instructional technology; (b) coordination be maintained with appropriate tri-service agencies for inter-service exchange; and (c) existing authoring aids, such as the Instructional Quality Inventory (IQI) and the Author Training Course, be modified if necessary and placed online as computer-based aids. A 24-item bibliography, a flow chart outlining Instructional Systems Development (ISD) design, and development tasks, and a report distribution list are provided. —Microfiche 97 cents, paper copy $2.15 plus shipping, as document ED 243 476.


This report reviews the background of the Instructional Systems Design (ISD) model, which is used to develop training for the Navy personnel; identifies problems in the ISD process and its management and implementation; and recommends methods of ISD improvement. The ISD model is described as a process originally developed to remind instructional development experts about steps needed to produce quality instruction, but subsequently implemented to help content specialists (who are relatively inexperienced in instructional design and development) build instruction. It is noted that ISD methods as used by non-experts are not successful because they lack detailed procedural guidance. Instructional engineering and management problems in implementing ISD are outlined and three alternative solutions to these problems are considered and rejected. Several recent research efforts are then summarized, including the instructional quality inventory (IQI), which provides quality assurance methods for the ISD; the development of guidelines for building more relevant criterion-referenced tests; and the initial development of computer-assisted training development. It is recommended that the Navy Education and Training Command (NAEDTRACOM) develop: (a) systematic methods for monitoring ISD implementation and the performance of ISD practitioners and managers; (b) training and professional development programs for these persons; and (c) automated aids for ISD. A 36-item bibliography and a report distribution list are provided. —Microfiche 97 cents, paper copy $2.15 plus shipping, as document ED 243 472.

Formative Evaluation in Instructional Design: Theory versus Practice. John A. Williams, Jr. Paper presented at the annual meeting of the Association for...
Information Resources, School of Education, Syracuse University, Syracuse, N.Y. 13210.


While the improvement of authoring systems and the development of knowledge-based rather than frame-oriented computer assisted instruction (CAI) systems are useful, a significant part of the courseware development problem lies in the inability of teachers (subject matter experts) to prepare material that promotes effective, efficient learning. Only those visible instructional events exemplified by "offline" independent study packages and online CAI sequences are available for review and criticism; classroom-based instruction receives no similar scrutiny. The problem does not lie solely in the development of educational technology; rather, it is more generic, existing in all instructional environments. The solution lies in developing an increased understanding of learning strategies, algorithms, and heuristics with this task falling largely on instructional system developers, and more specifically on instructional designers. The paper includes a description of courseware production problems at Athabasca University, which is a Canadian distance education institution, and recommendations for the development of a Canadian technology-based educational system. A summary of the paper in French is included.—Microfiche 97 cents, paper copy $2.15 plus shipping, as document ED 243 462.

The above documents may be ordered from the ERIC Document Reproduction Service (ERDS), P.O. Box 190, Arlington, Va. 22210. Please order by ED number, indicate the format desired (microfiche or paper copy), and include payment for the price listed plus shipping. Inquiries about ERIC may be addressed to the ERIC Clearinghouse on

Educational Communications and Technology, Dallas, January 1984. 44 pp.

The two phase study described in this report was designed to (a) investigate the appropriate evaluation literature and develop a formative evaluation model; (b) investigate the formative evaluation procedures utilized by Advanced Systems Incorporated (ASI), a successful producer of training materials; (c) develop a model based upon these procedures; and (d) compare and contrast the two resultant models for commonalities and differences. Results are reported from both a literature review of current formative evaluation procedures and theory and second-phase data gathering, which involved observation of evaluation procedures, extensive visitation, and an in-house survey of ASI employees involved in the product development process. An outline combines elements of education, military training, and industrial training into a single paradigm. Nine conclusions and recommendations for study applications and for further research are included. Sixtyseven references are listed and elements of the models discussed are illustrated with numerous charts.—Microfiche 97 cents plus shipping as document ED 243 439.


This guide focuses on appropriate techniques for the evaluation of electronic media educational programs. Such evaluation helps to provide the educator with the feedback that is missing whenever there is no direct contact with students, and also with information on the relative worth of an educational program. The four-part publication includes: (a) a statement of purposes and a description of radio and television as educational delivery systems, with emphasis on their use by the Cooperative Extension Service; (b) a brief review of evaluation systems as applied to education, with a discussion of some of the evaluation models that may be applied to electronic media educational delivery and a recap of several evaluations of educational television; (c) a practical approach to the evaluation of electronic media delivery of extension programs with a detailed discussion of each of the phases of evaluation, plus a section on