Contents

ARTICLES

Instructional Models for Tutoring: A Review
Linda A. Frey and Charles M. Reigeluth

Automated Instructional Development Using Personal Computers: Research Issues
Gregory Kearsley

The Delphi as a Job Analysis Tool
Jeffrey A. Cantor

Teaching the Mechanics of SAS with a Self-Instructional Manual: A Case Study
Barbara M. Florini and Jack Six

DEPARTMENTS

Book Reviews, edited by Allison Rosset

The Instructional Design Process
by Jerrold E. Kemp
Reviewed by Roberts A. Braden

Teachers and Machines: The Classroom Use of Technology Since 1920
by Larry Cuban
Reviewed by David Salisbury

Corporate Classrooms: The Learning Business
by Nell P. Eurih
Reviewed by Esther R. Sinofsky

Top Management Strategy: What it is and How to Make it Work
by Benjamin B. Tregoe and John W. Zimmerman
Reviewed by Dean R. Spitzer

ERIC Reports on ID, edited by Barbara B. Minor

Back Issues of JID
Instructional Models for Tutoring: A Review

Linda A. Frey
and
Charles M. Reigeluth
Syracuse University
School of Education
330 Huntington Hall
Syracuse, NY 13210

Abstract. This paper describes tutoring models and compares and contrasts the instructional strategies in those models that have been found effective by practitioners or researchers. The models reviewed include those of Literacy Volunteers, Laubach Literacy, Structured Tutoring, Programmed Tutoring, Peer-Mediated Instruction, and the Audio-Tutorial System. The review of these tutoring models reveals that there are different types of tutoring. Remedial versus first-time, mainline versus adjunct, and peer versus professional tutoring should be chosen on the basis of learner characteristics, setting, or learning task requirements. The tutoring strategies discussed are classified as organizational, delivery, and management strategies. The cost-effectiveness and adaptability of tutoring and conclusions concerning the further development of tutoring models are also discussed.

This paper describes, compares, and contrasts a variety of representative tutoring models that have been found to be effective by practitioners or researchers. Two important questions addressed herein are whether tutoring may require instructional strategies, and under what conditions tutoring should be chosen as an instructional mode. Commonalities and unique contributions of the tutoring models are identified to make it easier to integrate our current knowledge into optimal prescriptions for tutoring. It is hoped that this will both (a) provide the basis for further research and instructional theory development and (b) provide some further guidance for current practice.

Three questions, then, are of interest:
1. What is tutoring?
2. When should tutoring be used?
3. What strategies are unique to tutoring and not simply common to all good instruction?

Tutoring strategies were examined and classified as organizational, those dealing with the presentation components of the instruction; delivery, those dealing with materials, physical setting, or human contact; and management; those dealing with diagnosis, record keeping, prescriptions, and training of tutors. (Reigeluth, 1983).

Types of Tutoring

Tutoring is a kind of individualized instructional delivery mode. For the purpose of this review, we have used a simple but broad definition: instruction with one-to-one human interaction. We emphasize "human" and "interaction" because, unlike individualized instruction through print or machine media, a person is always the primary delivery mode. But unlike a lecture delivered by a person, tutoring always allows for continual, frequent response and feedback. "One-to-one" indicates non-group instruction in that a tutor is able frequently and rapidly to modify the instruction as it takes place, and to account for infinite combinations of learner and task characteristics or other constraints.

Tutoring can be done either by an expert (often called professional tutoring) or by an amateur (usually referred to as peer tutoring). Peer tutoring usually refers to arrangements in which the tutor and tutee have the same status but may not necessarily be the same age ("cross-age tutoring"). Peer tutoring may also take place between a learner and someone other than another student. School tutoring programs described in the literature often have parents, teacher-aids, or other paraprofessionals, paid or non-paid, who act as tutors. In addition to their lack of training, all of these amateurs differ from professionals in that they usually are supervised by someone.

For those students who are unsuccess-ful in traditional group instruction, tutoring is often remedial in nature. Thus, it is used as a follow-up to some other type of instruction. Tutoring may also be used for first-time, or non-remedial, instruction. The nature of the skill or subject matter may require tutoring. Most instrumental music instruction is first-time tutorial instruction. Instruction in microsurgery techniques may also require one-to-one instruction.

Further variation in the use of tutoring is suggested by the degree to which tutoring is utilized in the total instructional process. Tutoring can be a totally "stand-alone" (or mainline) process, as is usually the case in such non-school programs as Literacy Volunteers or Laubach Literacy International, or in on-the-job situations. However, tutoring can also be used in conjunction with other modes of instruction (adjunct tutoring). Tutoring is used to supplement such individualized self-instructional methods as the Personalized System of Instruction (Keller, 1968) and the Audio-Tutorial Method (Postlethwait, Novak & Murray, 1972). Tutoring programs may be institutional (i.e., sponsored by an institution) or non-institutional. Most research has been based on tutoring programs that are a part of traditional school settings, and most of those programs deal with remedial instruction in reading and mathematics.

Historical Background

A tutorial movement called the Bell-Lancaster system (sometimes referred to as the Monitorial or Madras System) gained popularity in England during the early 19th century (Thiagarajan, 1977). Andrew Bell, the superintendent of a school for orphans at Madras, India, devised a system using older children to...
teach other children in order to provide better instruction and discipline for the child. Professional educator Joseph Lancaster adapted Bell's idea to devise a monitoring system driven by economic rather than educational factors (Elby & Larsen, 1980) as a means to provide instruction for a growing student population in his school in England.

Lancaster's tutoring program was a complete instructional system. It included a systematically sequenced curriculum with procedures for tutors to follow, and it was inexpensive. Lancaster also developed a system of token economy using merit tickets (as reinforcement), which could be earned through both academic performance and behavior (Thaggarajan, 1977). The success of the system, however, depended upon precise, orderly management, which was not formerly possible with a single teacher and many students (Allen, 1976). Most recent tutoring models are based, at least in part, on the Bell-Lancaster system.

Recent Research and Practice

Tutoring Models (Practice)

The following is a brief description of some of the most prominent tutoring models (programs). It identifies the domain for which each was designed, including the kind of tutor, the kind of tutor, the skill area, the type of tutoring (e.g., first-time vs. remedial and adjunct vs. mainline), and the level of learning (remembering information vs. applying rules in new situations). Following this section is a description of the instructional strategies (organizational, delivery, and management) that are emphasized by each model. All of the characteristics of the models described below are compared in Table 1.

Literacy Volunteers of America, Inc. Literacy Volunteers of America (Colvin, 1980; Colvin & Root, 1981) is a national literacy tutoring program begun in Syracuse, New York. Its primary emphasis is on the tutoring of adults in reading and English as a second language through community-based, volunteer tutoring. Recently, however, Literacy Volunteers has had a program called the Teen Tutoring Program, in which high school students tutor younger students. Literacy Volunteers also sponsors tutoring in prisons.

Laubach Literacy International, Laubach Literacy International (Laubach, Kirk, & Laubach, 1981) is an organization founded by Frank C. Laubach that also enlist volunteer tutors to organize and administer local community-based literacy programs for teaching reading and writing skills to adults: both native speakers of English and speakers of other languages (Literacy Trainer Handbook, Macero & Lane, 1976).

Structured Tutoring. Grant Von Harrison (1972, 1975) has developed an instructional model called Structured Tutoring. In this model, tutors work with learners in a one-to-one situation. Harrison's model has been used in elementary schools for intergrade peer tutoring and for parent and paraprofessional tutoring in both reading and mathematics, often for remedial purposes.

Programmed Tutoring. Douglas Elison (1976) developed the Programmed Tutoring model in the 1960’s at Indiana University. The lesson content in Programmed Tutoring is highly structured and designed to lead the learner in small steps, much like programmed instruction. The instructional materials and teaching strategies are prescribed in enough detail that nonprofessional adults or students can be taught to use them quickly. Elison advocates the use of his tutoring program as an adjunct to academic classroom teaching rather than as a substitute for it (Elby & Larsen, 1980). Programmed tutoring has been used in institutional and non-institutional settings primarily for remedial instruction in reading and mathematics.

Peer-Mediated Instruction. Peter Rosenbaum (1973) developed a peer tutoring model based on the instructional program processes of computer-assisted instruction. In the PMI system, students work in pairs with one student designated as the “teacher” and one designated as the “student.” Peer-Mediated Instruction has been proven effective in teaching spelling at the elementary level, and in basic skills training with adults in American Telephone and Telegraph.

Personalized System of Instruction (Keller Plan). Fred Keller’s (1968) Personalized System of Instruction (PSI) is an instructional management system in which students work on individualized materials at their own pace in a classroom setting. Advanced students, or protors, provide one-to-one tutorial assistance to help students master the sequentially arranged materials. PSI originated at the college level, but has been used at all grade levels and most subject areas. It is also used in military and industrial training.

Audio-Tutorial System. The Audio-Tutorial System developed by Postlethwait, Novak, & Murray (1972) is an individually-paced, independent-study method using audio tapes. These tapes are tutorial conversations with the instructor. A live instructor is always available to give students needed tutorial assistance. Besides one-to-one instruction, the Audio-Tutorial System also utilizes large and small group instruction. The Audio-Tutorial system is used primarily in science education, but has been successfully applied in many other subject areas, at many age levels, and in both institutional and non-institutional settings.

Other Tutoring Models. Many other models for tutoring have been developed and used on a more local scale, such as Bramley (1979), Bright (1972), Ebersole & Dewitt (1972), Grabowski (1976), Lipsett (1975), Melarango (1972, 1976), Niedermeyer & Ellis (1972), Pope (1976), Rauch (1969), Thaggarajan (1976, 1977, 1978), U.S. Department of Health, Education, and Welfare, (1974a, 1974b), Verduin, Miller, & Greer (1977), and Wagner (1976). For the most part, each utilizes various components of the above described models.

Research

Most of the “structured” or “programmed” tutoring models, such as those developed by Elison, Harrison, and Rosenbaum, are supported by research. Other programs, such as the Literacy Volunteers of America and Laubach Literacy International programs, report
Table 1. A comparison of tutoring models

<table>
<thead>
<tr>
<th></th>
<th>LVA</th>
<th>Laubach</th>
<th>Structured Tutoring</th>
<th>Programmed Tutoring</th>
<th>PMI</th>
<th>PSI</th>
<th>Audio-Tutorial</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>STUDENTS:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>School Age</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>College</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adult</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td><strong>SETTING:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Informal</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>School</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>College</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Distance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Military</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bus/Ind.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TUTOR:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peer</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Professional</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SKILL AREA:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reading ESL*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reading ESL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reading Math</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reading Math</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spelling Basic Skills</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>All</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Science</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Most</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Others</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PURPOSE:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remedial</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>First-time</td>
<td>X(ESL)</td>
<td>X(ESL)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Adjunction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mainline</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td><strong>LEVEL OF LEARNING:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recall</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Application</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td><strong>TYPE OF STRATEGIES EMPHASIZED:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Organizational</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Generality</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Example</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Practice</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feedback</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Sequence</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Delivery</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Structured Materials</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Records</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Training</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rapport</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*English as a second language
anecdotal accounts of success.
Cohen, Kulik, and Kulik (1982) analyzed 65 studies dealing with elementary and secondary school peer-tutoring programs, and concluded that achievement was higher and that student attitudes toward instruction were more positive with peer tutoring than with conventional classroom instruction, but that the effect of tutoring on student self-concept was insignificant. Most of the studies also found higher achievement and positive attitude toward the subject matter for students who served as tutors. Cohen et al. (1982) found greater student achievement in studies in which (a) tutoring was a substitute for, rather than a supplement to, conventional classroom instruction, (b) tutors received training, (c) there was cross-age rather than same-age tutoring, (d) the tutoring was structured, (e) the tutoring was of short duration, and (f) the subject was mathematics.

In summary, tutoring practice and research indicate that tutoring is effective for a variety of settings, skill types, learning levels, and types of tutors and tutees. Tutoring may also be effective with other instructional formats (e.g., group instruction and self-instruction).

An Analysis of Strategy Components

Despite much diversity in the use of tutoring, many commonalities exist among effective tutoring models described in the literature. An examination of these models and of the research and practice in tutoring reveals that effective tutoring has the following common characteristics (or strategies): (a) a systematic arrangement of the subject matter to be taught, (b) specific, predetermined instructional strategies for the use of stimulus material, practice, and corrective feedback, (c) explicit management procedures, which include instructional prescription and records of student progress, (d) specific materials that facilitate the instruction and management processes, and (e) the training of tutors in instructional and management strategies.

Tutoring strategies described in the literature can be classified according to three main kinds of instructional strategies: organizational strategies, management strategies, and delivery strategies (Reigeluth & Merrill, 1979).

Organizational strategies refer both to micro strategies, which are strategies for teaching a single idea (such as explanation, example, practice, and feedback), and to macro strategies, which are strategies that relate to many ideas (such as selecting, sequencing, synthesizing, and systematically reviewing those ideas). Delivery strategies are those strategies for bringing the instruction to the learner and through which interaction with the learner can take place. They are primarily concerned with medium of instruction. Management strategies are those strategies for deciding which organizational or delivery strategies to use and when to use them in order to maximize learning.

Organizational Strategies

Most of the tutoring programs described in the literature deal with basic skills instruction in reading and mathematics and emphasize primarily practice and feedback strategies, which are micro-level organizational strategies. Ellson’s (1976) micro-level organizational strategies for his programmed tutoring model are based on Thorndike’s stimulus-response-feedback model. At the elemental level, tutoring consists of small units of teaching material that correspond to frames of programmed learning, according to Ellson. Ellson describes precise prompting sequences for practice and feedback, which Thiagarajan (1978) has used as a basis for specific instructions to tutors, called “tutoraids.”

Criteria are needed to determine when to use tutoring and which tutoring model to use.

Rosenbaum’s (1973) peer tutoring model emphasizes practice and feedback strategies at the micro level. Also, his “correction algorithm,” or corrective feedback process, is very similar to Ellson’s prompting sequence. Harrison (1978) indicates that tutoring should provide corrective and confirmatory feedback as well as appropriate examples, practice, and systematic review. Literacy Volunteers and Laubach Literacy International have less precisely described practice and feedback procedures, but do emphasize generality (or explanation), example, practice, feedback, and review. Literacy Volunteers does describe several practice and feedback strategies in flowchart form in its
Peer tutoring has a more positive effect on student achievement levels and attitudes than conventional classroom instruction.

Rosenbaum (1973) states that his Peer-Mediated Instruction can extend over all subject matters and ages, with the exception of complex industrial skills, or subject matter that cannot be operationally defined with enough specificity for peer tutors, such as foreign language instruction. The latter require the use of professional instructors for tutoring.

Summary and Conclusion

This review of tutoring reveals that there are different types of tutoring (remedial versus first-time, and mainline versus adjunct) and different types of tutors (peer and professional). These differences, as well as differences in target population, setting, skill area, and level of learning, suggest that different models are needed for different types of tutoring. Criteria are needed, therefore, to determine not only when to use tutoring, but when to use which tutoring model. To the extent that some (many) strategies are likely to be beneficial for all tutoring situations, it may be more useful to think in terms of different variations on a basic model rather than completely different models.

The review of tutoring models determined that there are several management strategies unique to tutoring, such as the selection, training, and monitoring of peer tutors, and the optimal physical setting. Some organizational strategies (which refer to the micro-level arrangement of the elements of the actual instruction) unique to tutoring, were also found: Ellison's "prompting sequences" and Rosenbaum's "correction algorithm" differ from what is normally thought of as programmed instruction because they must be created and modified based on the learner's response at the time of instruction. A closer examination of the presentation process is needed to determine whether the sequence of generality, examples, practice, and feedback differs from other modes of instruction. Preliminary investigation does suggest, however, that during the practice phase of instruction, strategies such as cueing, coaching,
The tutoring models should be chosen on the basis of learner characteristics, settings, and learning task requirements.

References


ECT Foundation
1986
Robert M. Gagné Award for Graduate Student Research in Instructional Development

$500

For the most significant contribution to the body of knowledge upon which instructional development is based. The Gagné Award competition is sponsored by the Division of Instructional Development of the Association for Educational Communications and Technology (AECT). A jury of scholars will select the best contribution. The award will be presented to the recipient at the AECT Annual Conference and COMMTEx International Exposition being held February 25-March 1, 1987, in Atlanta, Georgia. The winner will be presented a plaque and the cash award.

Eligibility

The work submitted must have been completed after December 31, 1983, while the award candidate was enrolled as a graduate student.

Nomination Procedure

You may nominate any individual (including yourself) for the Gagné Award. In order to nominate someone, send a copy (1) of the single piece of work (journal article, dissertation, etc.) for which they are being nominated. Send to:

Dr. Gary J. Anglin
University of Kentucky
Department of Curriculum and Instruction
136B Taylor Education Building
Lexington, KY 40506-0601

Deadline

Submissions must be postmarked no later than November 1, 1986.
Automated Instructional Development Using Personal Computers: Research Issues

Greg Kearsley
Courseware Incorporated

Abstract. This article discusses the design of systems for automating instructional development. A prototype automated Instructional Systems Design (ISD) system called the Courseware® Instructional Toolkit™ is described. The purpose of this system was to explore the extent to which ISD could be automated and the research issues associated with such automation. The major research issues identified were: (a) what aspects of ISD can be automated, (b) transfer of information between ISD tasks, (c) effects on ISD processes and outcomes, and (d) implications of automated systems for the skills of instructional developers.

Introduction

Instructional development is a very labor-intensive process. Because of this, it is very expensive, susceptible to individual differences in skill levels, and not especially reliable. Despite the emergence and widespread use of Instructional Systems Design (ISD) procedures (Logan, 1981; O’Neil, 1979), instructional development is still not a highly refined activity. There is a recognized need to improve the quality of ISD methods (Montague, Wulleck, & Ellis, 1983).

One potential solution to the problem is to automate instructional development procedures. Over the past decade, a number of research projects have examined this possibility. Braby and Kincaid (1981) developed a computer-based editing system that helped authors create Navy technical manuals. Brecke and Blaives (1981) described the CASDAT system that was intended to automate instructional analysis and design activities for technical training. Kincaid, Braby & Wulleck (1983) discuss computer aids for testing. O’Neal and O’Neal (1979) describe the Author Management System developed to automate project management in large-scale ISD projects. Merrill and Wood (1984) describe their Lesson Design System implemented on an Apple II microcomputer.

There are a number of different facets to automation and instruction (Kearsley & Seidel, 1985). A great deal of attention has been given to automating one particular aspect of instructional development, namely, the creation of computer-based instruction programs using authoring systems. Authoring systems allow computer-based programs to be developed without the need to know a computer programming language, and reduce the development time required (Kearsley 1982; 1984). There are literally hundreds of authoring systems available for personal computers, each one with different features and capabilities. Many instructional developers now use these systems to create computer-based instruction. However, authoring systems represent only the tip of the iceberg, even in the development of computer-based instruction. More time is required for the analysis, design, and evaluation of instruction than the actual creation of programs. To make more significant gains in reducing the time (and expense) associated with instructional development, it is necessary to automate these other tasks. While many existing authoring systems could serve as a good basis for more comprehensive automated development systems, at present they are limited to producing programs.

Another important aspect to automating instruction is the production of print and multimedia materials. Personal computers are now commonly used to create, edit and typeset textbooks and manuals as well as to generate graphics, slides, overhead transparencies, and animated sequences. As instructional developers become increasingly accustomed to using computers for media production tasks, it will become natural to use the same systems for analysis, design, and evaluation tasks.

To summarize, the instructional development field is being driven towards the use of computers for all phases of ISD by a number of factors including the desire for better quality results, the need for increased efficiency, and for convenience. Like other domains prior to the use of technology, there is little understanding in the ISD field of what kind of impact automation could have. The purpose of this article is to identify research issues in the development of automated ISD systems based on a prototype system called the Courseware® Instructional Toolkit™.

Architecture of the System

For the purpose of discussing the research issues involved in automating ISD, it is not necessary to extensively describe the Toolkit prototype. The prototype served primarily as a vehicle to identify and study the issues. It also served as a means to investigate the feasibility and resources required to develop a full scale system. On the other hand, the particular nature of the system undoubtedly influenced the kinds of issues raised. For this reason, the basic structure and characteristics of the system will be briefly described.

It should be emphasized that the Toolkit is intended for use in large scale ISD projects that involve a team of developers working together to produce a substantial amount of curriculum material (i.e., hundreds of lessons) under time pressure. In this environment, standardization and communication of results from one task to another is very important. While the Toolkit could be used by a solitary developer for a single course, most of the functions would probably not be worthwhile since the tasks could be completed faster and more easily by manual methods.

The Toolkit was implemented on an IBM PC/XT and consisted of four different types of software (see Figure 1).
The system manager integrated all of the other programs in the Toolkit and allowed them to exchange data. It also allowed us to store keystroke or command sequences (needed for macro capabilities—described below) and provided "windowing" capabilities across all programs. We used DESQ from Quarterdeck Office Systems, although there are a number of other similar programs now available for the IBM-PC.

Applications software includes commercially available programs for word processing, database management, spreadsheets, project management, statistical analysis, telecommunications, etc. Authoring software includes any commercially available authoring language or system, as well as programs for graphics creation, typesetting, and slide production. Custom designed software refers to programs that had to be written in a programming language to accommodate unique requirements that could not be met by the commercially available applications or authoring programs. Figure 2 shows the specific ISD functions that we explored in the prototype. Functions with an asterisk were actually implemented; those without were designed only on paper. These particular functions were selected for one of three reasons: (a) they represent commonly performed ISD tasks, (b) they represent ISD tasks typically performed poorly, or (c) they represent ISD tasks that are seldom performed (but should be).

Figure 1. Instructional Toolkit System Architecture

Figure 2. ISD Functions Implemented in Toolkit
Associated with each ISD function is a standard set of components: overviews, macros, shells, examples, job aids, and helps. The overview explains the nature and rationale of the ISD task and relates it to other ISD tasks. Although the overviews in the prototype were for reference only, they could be elaborated into full interactive tutorials that teach the ISD task to new developers. Macros represent the automated procedures and consist of stored command sequences appropriate for the applications or authoring program being used for that ISD function.

Shells represent an empty template or outline that can be completed (on-line or off-line) to accomplish the particular ISD task involved. For example, the shell for the problem analysis function, is the outline of a problem analysis document to be filled in using a word processing program. The shell for the task hierarchy function is a blank set of boxes to be filled in using a graphics program. The shell for a cost/benefit function is an empty spreadsheet template.

Examples are shells that have been completed for specific instructional projects. They serve as models or illustrations of what the task looks like when completed. Job aids are descriptions of the steps or procedures to be followed to complete an ISD task. In some cases, the job aid is simply a shell with a listing of the questions to be answered to complete the task. Helps explain how to use the current function. This could be an explanation of what the macros do or a specific step in the current task.

Examples of Use

A few examples will help clarify how a developer uses the system. Figure 3 shows a schematic of how lesson specifications are written. The lesson specification function in the Toolkit prototype used a word processing program as its basis. After selecting the lesson specification function, the developer can choose to see an overview, example, shell or job aid. There are three ways the designer could complete the task: (a) select a shell and fill in the appropriate information under the headings provided, (b) select an example and use this as a model to complete the shell, or (c) select the job aid and answer the questions to fill in the shell. Because each component can have its own window, it is possible to keep the example or job aid on one part of the screen while filling in the shell.

Once the specification has been completed, it can be printed out for review. If the developer is working with others on a network, the specification could be viewed on the screen. Since communications can be brought up in its own window at the same time a specification is being reviewed, it would be possible for two developers to have an on-line conversation about the specification. Because we did not have a net-worked system, we were unable to explore this interesting possibility.

Many of the analysis, development, and evaluation functions use word processing programs and work in a fashion similar to the specification function (e.g., problem analysis, job/task analysis, instructor training, test construction). Other functions, such as task validation, syllabus development, media selection, course evaluations, and student records use database management programs.

Figure 4 shows the sequence of steps in a syllabus development task. To construct a syllabus, the developer begins by transferring objectives to the database shell from an objectives hierarchy or task listing already created. The shell has columns corresponding to the critical parameters that will be used to organize the objectives into units or lessons (e.g., prerequisites, type of learning, duration). The developer must specify the parameters and a rule for combining objectives based upon the parameters (e.g., hands-on objectives at end of unit, maximum of 20 minutes per unit).

Once these parameters have been indicated for each objective, the program sorts the objectives into units based upon the rule provided. The result is then displayed to the developer who can change the rule or parameters and rerun the program. When the designer is satisfied with the syllabus, it can be printed out.

All of the operations (specifying parameters and rules, sorting, and printing) are handled by macros. The syllabus development function also includes the other standard Toolkit com-
ponents (overview, examples, job aid, and helps). Using this function, a large syllabus encompassing thousands of objectives and hundreds of units can be created in a matter of hours compared to days by manual methods. Furthermore, the process is repeatable—given the same parameter values and rules, the program produces the same syllabus each time.

The development of questionnaires for course evaluations is another function that uses a database management program. To compose a questionnaire, the developer simply selects the items to be included. The items can be open-ended questions as well as multiple choice or ratings. New items can be added by bringing up a word processing program in another window. After the questionnaires are completed and the responses entered, a statistical analysis program could be used to analyze the results and keep these in the course evaluation database.

Project costing (a project management task) is an example of a function that uses a spreadsheet program. Other functions that use spreadsheet programs as their basis are training requirements estimation and cost/benefits analysis. The shell for this program is a template that requires the developer to fill in values for the estimated number of objectives or course duration, type of media involved, complexity of content, distance between customer and project staff, and other critical project costing parameters. Once these values have been filled in, the program uses a matrix of historical data to compute the estimated number of person-days required for different job categories and costs based upon salary and overhead rates.

The developer can model different assumptions about course size, media, and project staffing to produce a range of cost estimates. Using this function, project costing can take a few minutes to complete instead of hours using a calculator. More importantly, because the function uses cumulative historical data, the estimates are likely to be more accurate than ones made on the basis of a single individual's experience.

Cost/benefits analysis is another example of a function that uses a spreadsheet program. The developer first completes the system and development costs and then estimates the utilization of the program. The program then calculates the cost per student hour. If any of the entries are changed, the calculation is immediately redone. This allows the developer to explore different assumptions about costs or utilization.

Issues
Putting aside the obvious practical benefits alluded to in the previous section (i.e., standardization, time savings, increased reliability, more accuracy), the Toolkit prototype raises some interesting research questions about
automated instructional development. They are as follows:

1. **What aspects of ISD can/cannot be automated?**

   This is a classic issue in the field of human factors engineering that deals routinely with performance limitations in human-machine systems. It is also an important question in the field of artificial intelligence, which deals with the design of computer programs capable of "intelligent" actions. Both of these perspectives are relevant to the issue of what aspects of ISD can/cannot be automated.

   Most ISD tasks can be broken down into a relatively simple clerical component and a more complex judgemental component. For example, in lesson specification, certain questions must be asked and the answers obtained (e.g., "What is content for this objective?"; "How could you illustrate this concept visually?", etc.). This is a simple clerical task. The answers must then be translated into a specification for a definition, example, test question, review, and so on. This step requires judgement. In syllabus development, the judgement comes in identifying the parameters of each objective and the grouping rules; the sorting into units is clerical in nature. In project costing, the judgements are made in estimating the size and complexity of the project; the generation of the actual cost estimates is more calculation.

   The clerical components of a task are relatively easy to automate. They consist of procedures with a fixed set of steps, formulas, or decision rules. On the other hand, the judgemental components are more difficult to automate. They depend upon context-dependent rules, very detailed discriminations, or heuristic strategies. They are often difficult to state explicitly.

   The clerical component is very susceptible to error. A general finding of human factors research is that most people produce high failure rates in repetitive tasks due to fatigue, boredom, and loss of attention. For this reason, automation of the clerical component is usually a good idea.

   Automation of the judgemental component depends heavily on artificial intelligence approaches (including expert systems). For example, consider the task of making decisions about the successful use of computer-based training. Judgements about the suitability of a course for computers or the appropriate kind of system to use depend upon a series of inter-related rules derived from actual computer-based training projects. These rules can be put in the form of an expert system that provides training managers with advice for specific instructional applications (Keatsley, 1985).

   The Toolkit prototype focused on the clerical component of ISD tasks. More sophisticated software and a much more detailed understanding of the judgemental process in instructional development would be needed to go beyond this level.

2. **What are the relationships among different ISD tasks?**

   The Toolkit was designed as an integrated system that allowed the results of one ISD task to be passed forward or backward in time to another ISD task. This raises the issue of the usual or possible data transfer paths among different ISD tasks. When ISD is done manually, the flow from one task to another is

---

**Figure 5. Predicted Relationships Between ISD Tasks**

<table>
<thead>
<tr>
<th>Problem Analysis</th>
<th>Job/Task Analysis</th>
<th>Task Validation</th>
<th>Objective Hierarchies</th>
<th>Lesson Specifications</th>
<th>Media Selection</th>
<th>Syllabus Development</th>
<th>Prototyping</th>
<th>Storyboarding</th>
<th>Authoring</th>
<th>Test Construction</th>
<th>Project Management</th>
<th>Instructor Training</th>
<th>Training Requirements</th>
<th>Course Evaluations</th>
<th>Cost/Benefits</th>
<th>Bullet-proofing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
seldom identified. Furthermore, since it typically takes place over a period of days or weeks, it would be difficult to track. In an automated ISD system, however, the transfer of data between tasks can occur in a matter of seconds. Since we would like to automate as many transfers from task to task as possible, these relationships are of considerable importance in an automated system.

Figure 5 is a theoretical plot of the frequency of transfers between the 17 ISD tasks in the Toolkit prototype. Each cell indicates the likelihood of an interaction between one task and another. The darker the cell at the intersection of any two tasks, the greater the degree of interaction predicted between them. For example, some transfers are routine (e.g., moving objectives to syllabus development or test construction, lesson specifications to storyboarding, evaluation data to lesson specifications) while others are rare. Note that the plot is asymmetrical; transfers have directionality. Most interactions are predicted to occur in a forward direction (the top left diagonal of the matrix) rather than backward (bottom right diagonal). To validate this plot empirically, it would be necessary to count the frequency of task interactions in the Toolkit across time.

Another important part of this data transfer issue is whether there are "optimal" ISD paths for a given instructional project and goals (e.g., least time/cost, best quality outcomes, smallest project team). By conceptualizing ISD tasks as nodes in a network connected by their transfer paths, it is possible to examine optimal paths using various network analysis methods. The network will usually be different for each project because projects include different subsets of ISD tasks and start/stop at different tasks.

3. How will automation affect the process and outcomes of ISD?

There are many ways that automation could change how ISD is conducted and the results of the process. For example, the availability of ISD functions in an automated system could encourage their use. ISD tasks that are tedious or difficult to do manually are likely to be skipped. In an automated system, such tasks may be carried out because they are less tedious or difficult. Thus, one of the possible effects of automated ISD systems is that more ISD tasks would be completed for a given project.

In an open architecture system like the Toolkit, it is possible that a developer might have a choice of two or more functions for a given ISD task. For example, in the prototype, there were two levels of media selection: a macro analysis for making judgments based upon the attributes of an entire course, and a micro analysis based upon the attributes for every objective of a course. In this situation, the developer must select which function to use (or try both). This requires the developer to make a judgement about the relative merits of one ISD method over another for a given project. Will developers make the best choice, and if they make inappropriate choices, how and when can this be detected?

This raises a fundamental point—how can we measure the quality of ISD outcomes in order to detect the effects of automation? The ultimate test of ISD quality is the extent to which the instruction works, i.e., helps people learn. It would seem a simple matter to assess the effectiveness of ISD methods by measuring differences in job performance or achievement. Of course, there are too many intervening variables to make this possible. A more feasible test for automated ISD is to compare the results of a particular task done manually and via the system. In other words, if one developer completes the task manually and another developer (of matched experience) completes the task using the system, what are the differences in the resulting outcomes? Do they differ in completeness, or correctness? Is a syllabus or lesson produced by a developer using the Toolkit as good, better or worse than one produced by the same developer without the system?

From our work with the Toolkit prototype, it is clear that one of the major effects of an automated ISD system is on the time required to complete ISD tasks. The use of the Toolkit produced time savings in the order of 30 percent for some tasks. Furthermore, the delay between finishing one task and beginning another can be minimal. On the other hand, some tasks took longer because developers experimented with different alternatives or variations. In addition, when there were problems with the system or the hardware, no work was accomplished. It is apparent that the use of automated ISD systems will change the timescale of instructional development, although not necessarily reduce the time required.

4. What are the implications of automated ISD systems for the skill levels of instructional developers?

One enduring misconception about technology is that its major purpose is to substitute for human abilities and replace people. In fact, the rationale for using technology is always to improve productivity by augmenting or extending human abilities. In the case of automated ISD, we are interested in creating tools that allow developers to accomplish more efficiently or effective instructional development. The system should compensate for the lack of experience of a new developer. The system should also permit experienced developers to leverage their skills and eliminate time wasted in routine aspects of ISD tasks.

However, the effective use of an automated system requires mastery of the system. In the case of the Toolkit prototype, it was necessary for developers to attain a general level of computer literacy as well as obtain some familiarity with the specific application programs used (i.e., word processing, database management, spreadsheets, etc.). A considerable amount of time was required to learn how to use all of the features and capabilities of the system. For developers to take advantage of the system and use it productively, this time

Automated ISD prototypes like the Toolkit help us explore new ISD capabilities that are not possible without the computer.
One of the new skills needed by developers is the ability to select one approach versus another for a given training project. Most traditional training of instructional developers tends to focus on the use of a particular ISD methodology. However, in an automated ISD system such as the Toolkit, it may be necessary to choose the most appropriate ISD function from a selection of possibilities. This suggests that developers will need a more comprehensive understanding of ISD in order to use an automated system effectively.

The implication of these observations is that an automated ISD system may be more valuable to an experienced developer than a novice. This constitutes a hypothesis that could be tested in further research.

Conclusions

Because of time, budget, and equipment limitations, it has not yet been possible to extensively field test the Toolkit prototype in actual ISD projects. Various parts of the Toolkit have been used in projects and have led to the observations and hypotheses described in this article. The prototype has demonstrated that an automated ISD system provides a rich environment for conducting research on instructional development methods.

One intriguing aspect of automating ISD that was not explored in the prototype is the development of functions for tasks that cannot be done well by manual means. For example, Figure 2 lists functions for prototype development and bullet-proofing. The prototype development task involves creating a sample of a course or lesson before development starts. This task is important to avoid misunderstandings about the format or organization of materials. However, because making a prototype is time consuming and expensive, this step is often skipped (with dire consequences). However, there are programs available that make it possible to easily create an electronic "mock-up" of a printed, audiovisual, video or computer lesson that could be used as a prototype.

Bullet-proofing involves testing for all the content and logical flaws in instructional materials, especially interactive programs. To test a lesson for all flaws, it would be necessary to generate every possible student response. It is easy to write a computer program than will generate every possible response at random. It is also possible to write a program that simulates a student and makes likely responses for each instructional sequence based upon typical learner behavior (including misconceptions and errors). Traditional bullet-proofing technique depends upon field testing that is very tedious and time-consuming. Furthermore, traditional bullet-proofing methods are not very comprehensive and identify only a small fraction of problems.

The point of these two examples is that automated ISD could help us explore new ISD capabilities that are not really possible without the computer. The use of computers has already led to the development of new theory in many areas of instruction (Kearsley, Hunter & Seidel, 1983). There is no reason why it couldn't also have this effect in the instructional development area.

The major stumbling block in the development of automated ISD systems is our lack of detailed understanding of the instructional development process. In order to write a program to carry out an ISD procedure, it must be possible to describe it explicitly. However, much ISD methodology is based upon intuitive judgements and tacit knowledge. In order to automate such processes, this intuition and tacit knowledge must be made explicit. One of the valuable outcomes of research in automating ISD is likely to be a more complete understanding of the creative steps in instructional development. It seems ironic that in creating programs to mimic human behavior, we should learn more about it.

Acknowledgement: The Instructional Toolkit prototype was the joint effort of many employees of Courseware Inc. In particular, Nile Gardner, Diana Pickens, Susan Lindgren, and Kathy Rose made significant contributions to its design and development.

References


The Delphi as a Job Analysis Tool

Jeffrey A. Cantor
DDL OMNI Engineering
McLean, Virginia

Abstract. The objective of this project was to revalidate a course of instruction. Traditional approaches, methodologies and resources for validating instruction were considered. Technical documentation was reviewed and subject matter experts were consulted. The findings indicated a discrepancy between the existing instruction and the needs and goals of the training population. This scenario is probably familiar to many instructional designers, however, the approach selected to redesign the course is probably not. Job and task analysis (JTA) methods are amply described in the instructional development literature. These methods include documentation review, observations of the workplace, interviews of incumbents and subject matter experts and various kinds of surveys. The methods work well in areas where the job is well defined and stable and where changes in job requirements are predictable.

Recently I was charged with the responsibility of analyzing a course of study for preparation of commanding officers (COs) for the Navy's submarine force. As I gathered information relevant to the course, I began to realize that a discrepancy existed between the content of the present course and the training needs of the prospective submarine commanding officer. The existing course was a combination of portions of other technician training courses in the various technical job areas of the ship. As a result, the course was more detailed than was necessary in some aspects, and in others, omitted information and skills which would be appropriate and necessary for the CO. Therefore, a method capable of assessing needed information, appropriate level of instructional detail, and best method of presentation was required. Due to the nature of the CO's job, an assessment of future command needs based on current thought and reflection, was also required. The demands which would be placed on the schedule and workload of the CO prohibited using a group or committee information gathering process (Finch & Crunkleton, 1979; Adams, 1975). However, information from the CO incumbents was essential as they had first-hand experience about the information, skills, forms, and presentation styles required for the course. An approach which would provide for future forecasting needs and which would allow for participation without travel and time expense was indicated. Such an approach was the Delphi Process.

The Delphi Procedure

Review of the Literature

Weisbrod and Swenson (1980) describe the Delphi technique as a sequence of related procedures for eliciting and refining information and opinions obtained from a group. As a process of sequential individual interrogations centered around a questionnaire which is interspersed with both information and opinions, the Delphi capitalizes on the concept that "several heads are better than one" in arriving at subjective conjectures about the future (Dalkey & Helmer, 1963). "Experts...will make conjectures based upon rational judgment and shared information rather than merely guessing, and will separate hope from likelihood in the process (Dalkey & Helmer, 1963, p.9)."

The term "expert" can be applied to anyone who can contribute relevant information and opinions. However, according to literature generated by the Rand Corporation (Dalkey, 1969) and further supported by Pill (1971), the term "expert" generally refers to "highly educated and experienced specialists in particular subject disciplines" (p. 6). Implicit in this context, the Delphi procedure could be employed in almost any social situation to gain meaningful information concerning opinions about events—past, present, or future—and then quantifying and ordering that subjective information. Pill (1971) stated with respect to the question of utilization of the procedure in a decision-making process: "Perhaps the first step in being able to make reasonable decisions about the future, is to understand the present, and by implication the past" (p. 7).

Dalkey (1969) describes the Delphi procedure as having three desirable features: anonymity, controlled feedback, and statistical group response. The feature of anonymity is established through the use of questionnaires or other impersonal communications channels such as computers. This feature reduces the effect of dominant individuals. Irrelevant information is reduced through controlled feedback—conducting the exercise in a sequence of rounds between which a summary of the results of the previous round are communicated to the participants. Statistical group response is a device to assure that the opinion of each group member is reflected in the final response. The use of statistical definition of the group response is a means of reducing group pressure for conformity. However, at the end of the exercise there still may be a significant appeal in individual opinions. Based on this analysis, the Delphi procedure would serve as a basis for eliciting responses needed to quantify information and opinions, which would serve as a basis for further decision-making or analysis.

The Delphi process has been used in the field of training education. Weaver (1971) and Winstead & Hobson (1971) describe the use of the process in establishing educational priorities and institutional goals, and in forecasting the future for planning purposes. Finch and Crunkleton (1979) view the Delphi as a useful tool when curriculum planners
desire to reach consensus regarding the content of a particular curriculum. They offer the technique as useful in enabling experts to speculate individually and then reach consensus regarding the content necessary to prepare workers, even in areas where no workers presently exist. Lester and West (1979) describe the use of the Delphi technique to identify and classify the critical tasks of the educational information specialist. The Delphi process appeared to be appropriate for validating the Navy CO course content.

Method
A group of 14 active submarine commanding officers who would participate in the Delphi process, was identified through nomination and selection by the Navy Command, based upon their patrol and command experience. Prior to constructing a questionnaire suitable for the process, a series of three planning interview meetings were held with several former commanding officers (not part of above sample) for the purposes of identifying overall major areas of concern which would be offered as possible starting points for round one of the Delphi process.

Round one of the Delphi process requested the participants to reflect on their experiences as commanding officers and comment on the appropriateness of topics suggested for inclusion in the course. The participants were also requested to suggest all of the possible topics which should be included in the course. Additional requested information included the level of detail at which the topic should be taught, the most appropriate presentation methods for each topic, and the rationale for the choices. Figure 1 presents the format for the round one Delphi instrument.

The participants were given one week in which to respond. They were specifically asked to consider their experiences in deciding topics for inclusion and exclusion. As a result of the data analysis, 36 topic areas were identified for inclusion in the course. The data indicated a dichotomy of opinion as to appropriate level of detail for certain technical topics. This was a predictable outcome. The rationale for choice of level of detail would become important data in round two. These data were summarized and compiled into a round two instrument.

In round two of the Delphi process, each of the 14 participants was requested to review his initial responses to each topic. The participants' original instruments (round one) were copied and returned to them with their round two instruments. They were then requested to review the group responses to each topic. Each participant was then requested to react to the group's response and to his original response with respect to each of the variables (Include, Not Include, Emphasis, and Method of Instruction). Each participant was also requested to review the additional topics suggested and to support all of his remarks with a current rationale. Figure 2 represents the format for the Delphi round two questionnaire.

As a result of round two of the Delphi process, the same 36 topic areas were identified for inclusion with reactions indicating an increased consensus for each topic area across each of the variables. These data were then compiled for round three—the final round of the Delphi process.

<table>
<thead>
<tr>
<th>List I (Emphasis)</th>
<th>List II (Teaching Methodology)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. Concepts</td>
<td>1. Lecture</td>
</tr>
<tr>
<td>b. General Overview</td>
<td>2. Self-Study</td>
</tr>
<tr>
<td>c. Block Diagram</td>
<td>3. Guided Seminar</td>
</tr>
<tr>
<td>d. Detailed Theory</td>
<td>4. Group Study</td>
</tr>
<tr>
<td>e. Procedural Discipline</td>
<td>5. Demonstration</td>
</tr>
<tr>
<td>f. General Operating Guidelines</td>
<td>6. Lab Drill</td>
</tr>
<tr>
<td>g. Sequential Operation</td>
<td>7. Other</td>
</tr>
<tr>
<td>h. Effect or Casualities</td>
<td></td>
</tr>
<tr>
<td>i. Information Flow Path</td>
<td></td>
</tr>
<tr>
<td>j. Organizational Responsibilities</td>
<td></td>
</tr>
<tr>
<td>k. Maintenance Policy</td>
<td></td>
</tr>
<tr>
<td>l. Black Box Functions</td>
<td></td>
</tr>
<tr>
<td>m. Scheduling</td>
<td></td>
</tr>
<tr>
<td>n. Detailed Operating Procedures</td>
<td></td>
</tr>
<tr>
<td>o. Detailed Maintenance Procedures</td>
<td></td>
</tr>
<tr>
<td>p. Security</td>
<td></td>
</tr>
<tr>
<td>q. Safety</td>
<td></td>
</tr>
<tr>
<td>r. Other</td>
<td></td>
</tr>
</tbody>
</table>

SAMPLE

1. EM Log

Include [x] Not Include [ ]

Emphasis [ ]

Method of Instr. [ ]

ALL PCO/PEO should be familiar with the EM Log and use of its output. SSBN unique applications of the log should be emphaszized.

Figure 1. Delphi Round 1 Questionnaire Item.
In round three of the process, the participants were again requested to review their round two responses to each topic area as well as the group responses to that topic area. They were again requested to react to both the group responses as well as their responses for each topic with respect to each of the variables and to support their choices with a rationale. In each case a rationale was requested whether or not changes were made in their choices. Figure 3 represents the format for Delphi round three.

The results of round three were tabulated. At the conclusion of round three, two respondents changed their responses to the technical topic areas and conformed to the majority of the group, thus indicating a need for only an overview presentation on the topic. Data which resulted were then analyzed in order to produce an outline of the curriculum as suggested by the participants.

Results

As a result of this three-round Delphi process, the revised course contained 36 topic areas. Each topic area was identified with appropriate emphasis and teaching methodology. The revised course reflected a less detailed equipment knowledge emphasis and a more conceptual overview of equipment systems. The revised course also emphasized concentration in areas of departmental administration previously not covered. The previous course was developed largely from technical documentation and information provided by technicians, rather than from information provided by actual job incumbents. The Delphi process provided this input. The Delphi process has now become a standard practice for course definition and validation in the Navy submarine training program.

Discussion

The Delphi process has proven to be a useful tool for job analysis. As employed in this case, the Delphi process provides a means for course validation, a useful exercise often overlooked in many training organizations.

Participants in this process indicated a desire to participate again in the future. The Delphi methodology enabled me to gather data from critical participants without the burden of time, travel and expense. It also provided a means of fostering group thinking without pressure from peers and superior officers.

The Delphi methodology provided a means of obtaining information from knowledgeable people who are currently performing the job under study. This group can most adequately predict those skills and areas of knowledge which will be necessary in the foreseeable future. The process also permitted time and an appropriate atmosphere for each participant to reflect on the responses of his

---

### Table

<table>
<thead>
<tr>
<th>List I (Emphasis)</th>
<th>List II (Teaching Methodology)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Concept</td>
<td>12. Lecture</td>
</tr>
<tr>
<td>2. General Overview</td>
<td>13. Guided Seminar</td>
</tr>
<tr>
<td>4. Detailed Theory</td>
<td>15. Demonstration</td>
</tr>
<tr>
<td>5. Procedural Discipline</td>
<td>16. Lab Drill</td>
</tr>
<tr>
<td>6. General Operating Guidelines</td>
<td>17. Other</td>
</tr>
<tr>
<td>7. Effect of Casualties</td>
<td></td>
</tr>
<tr>
<td>8. Information Flow Path</td>
<td></td>
</tr>
<tr>
<td>9. Organizational Responsibilities</td>
<td></td>
</tr>
<tr>
<td>10. Maintenance Policy</td>
<td></td>
</tr>
<tr>
<td>11. Black Box Functions</td>
<td></td>
</tr>
<tr>
<td>12. Scheduling</td>
<td></td>
</tr>
<tr>
<td>13. Detailed Operating Procedures</td>
<td></td>
</tr>
<tr>
<td>14. Detailed Maintenance Procedures</td>
<td></td>
</tr>
<tr>
<td>15. Security</td>
<td></td>
</tr>
<tr>
<td>16. Safety</td>
<td></td>
</tr>
<tr>
<td>17. Other</td>
<td></td>
</tr>
</tbody>
</table>

The results offered by some of the respondents are as follows:

1. Needs to know overall system and interrelationships of basic building blocks such as sub-systems, components, maintenance, scheduling and casualty procedures (8 respondents)

Please review your response to item #1 in light of the above information. You may maintain your original position or change it. In either case, please provide justification for your action. Below is a space and format provided for your next response.

---

**Table Example**

<table>
<thead>
<tr>
<th>List I (Emphasis)</th>
<th>List II (Teaching Methodology)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. Concept</td>
<td>1. Lecture</td>
</tr>
<tr>
<td>b. General Overview</td>
<td>2. Self-Study</td>
</tr>
<tr>
<td>c. Block Diagram</td>
<td>3. Guided Seminar</td>
</tr>
<tr>
<td>d. Detailed Theory</td>
<td>4. Group Study</td>
</tr>
<tr>
<td>e. Procedural Discipline</td>
<td>5. Demonstration</td>
</tr>
<tr>
<td>f. General Operating Guidelines</td>
<td>6. Lab Drill</td>
</tr>
<tr>
<td>g. Effect of Casualties</td>
<td></td>
</tr>
<tr>
<td>h. Information Flow Path</td>
<td></td>
</tr>
<tr>
<td>i. Organizational Responsibilities</td>
<td></td>
</tr>
<tr>
<td>j. Maintenance Policy</td>
<td></td>
</tr>
<tr>
<td>k. Black Box Functions</td>
<td></td>
</tr>
<tr>
<td>l. Scheduling</td>
<td></td>
</tr>
<tr>
<td>m. Detailed Operating Procedures</td>
<td></td>
</tr>
<tr>
<td>n. Detailed Maintenance Procedures</td>
<td></td>
</tr>
<tr>
<td>o. Security</td>
<td></td>
</tr>
<tr>
<td>p. Safety</td>
<td></td>
</tr>
<tr>
<td>q. Other</td>
<td></td>
</tr>
</tbody>
</table>

**Sample**

1. EM Log

   **Include** ✔️ **Not include** ✗

   **Emphasis** 4th, k

   **Method of Instr.** 3

   **ALL PRO/PRO should be well versed with the EM log and use of its output. SBN unique applications of the log should be emphasized.**

---

**Figure 2. Delphi Round 2 Questionnaire Item.**
1. EM LOG

To include  12
Not to include  1

<table>
<thead>
<tr>
<th>List I (Emphasis)</th>
<th>List II (Teaching Methodology)</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 a. Concepts</td>
<td>12. Lecture</td>
</tr>
<tr>
<td>12 b. General Overview</td>
<td>1.2. Self-Study</td>
</tr>
<tr>
<td>0 c. Block Diagram</td>
<td>03. Guided Seminar</td>
</tr>
<tr>
<td>1 d. Detailed Theory</td>
<td>14. Group Study</td>
</tr>
<tr>
<td>0 e. Procedures</td>
<td>05. Demonstration</td>
</tr>
<tr>
<td>0 f. General Operating Guidelines</td>
<td>06. Lab Drill</td>
</tr>
<tr>
<td>0 g. Sequential Operation</td>
<td>07. Other</td>
</tr>
<tr>
<td>0 h. Effect of Casualties</td>
<td></td>
</tr>
<tr>
<td>0 i. Information Flow Path</td>
<td></td>
</tr>
<tr>
<td>0 j. Organizational Responsibilities</td>
<td></td>
</tr>
<tr>
<td>1 k. Maintenance Policy</td>
<td></td>
</tr>
<tr>
<td>0 l. Black Box Functions</td>
<td></td>
</tr>
<tr>
<td>0 m. Scheduling</td>
<td></td>
</tr>
<tr>
<td>0 n. Detailed Operating Procedures</td>
<td></td>
</tr>
<tr>
<td>0 o. Detailed Maintenance Procedures</td>
<td></td>
</tr>
<tr>
<td>1 p. Security</td>
<td></td>
</tr>
<tr>
<td>0 q. Safety</td>
<td></td>
</tr>
<tr>
<td>1 r. Other</td>
<td></td>
</tr>
</tbody>
</table>

Include  Not Include  Emphasis  Teaching Method  Responses
\[ \checkmark \]  [b]  [1]  
\[ \checkmark \]  [b]  [4]

Describe use of EM Log output (2 respondents).

Emphasis SSBN unique applications (1 respondent).

Remarks/Justification

Agree that an overview of SSBN unique applications should be included; however, subject does not require "emphasis."

Figure 3. Delphi Round 3 Questionnaire Item.

References


Weatherman, R., & Swanson, K. (1980). Delphi technique, Educational Futures.


1986, VOL. 9, NO. 1
Teaching the Mechanics of SAS with a Self-instructional Manual: A Case Study

Barbara M. Florini
and
Jack Six
Center for Instructional Development
113 College Place
Syracuse University
Syracuse, New York;

Abstract. With the increasing demand for skill and knowledge in the operation and application of computer technology, colleges and universities are being pressured by their students to provide effective and efficient instructional programs in this area. The following case study describes how one university, through its instructional development agency, was able to develop, evaluate, and refine a self-instructional manual that provides management students with the basic skills required for using the Statistical Analysis System (SAS), a major statistical program available for mainframe computers. The successful use of this manual with several hundred students over the course of five semesters and three summer sessions suggests that the manual may be a useful model for institutions with similar needs.

This paper describes the development, use and evaluation of a self-instructional manual to help solve a persistent computer-related problem at a university. Each semester large numbers of students from various disciplines are required to develop basic skills in using the Statistical Analysis System (SAS), a statistical analysis and reporting program that is available on the campus mainframe computer. Because of the wide availability of SAS on large computing systems, it is unlikely that the problems students face in learning to use SAS are unique to one campus. This case study is presented in the hope that others might benefit from the experience described below.

Following a description of the initial situation, the following topics are discussed:
- the development process leading to the manual.
- the manual itself.
- the introduction of the manual into the instruction setting.
- the evaluation plan and the results of the evaluation.

The paper concludes with several general observations and recommendations.

Description of the Situation

For some time, the use of SAS by students has taxed the academic computing consultant resources on campus. The general feeling was that students new to computing were having difficulty learning to use sophisticated applications programs like SAS in spite of classroom instruction and the availability of a consultant's room in the computing center. As a result, the Center for Instructional Development at the University was asked by a senior administrator with oversight for campus computing services to devise some efficient and effective means to help students learn SAS. Given this request, the growing demand on computing resources, and the large numbers of students needing to learn SAS, the project was given high priority within the Center.

Problem Conformation

In an effort to find out if the problem was amenable to an instructional solution, a series of discussions was initiated with members of the computing center staff and faculty having prior experience teaching SAS. These discussions confirmed that large numbers of students routinely sought help in running even simple SAS jobs despite classroom instruction, ready availability of reference documents, and on-line instructions for using SAS.

Problem Analysis

The next step was to identify the major elements of the problem. This was done through discussions with representatives from the Academic Computing Center, faculty members, and students. The following five major factors emerged.

1. SAS itself. The Statistical Analysis System is a powerful, complex computer program. It enables users to analyze data by means of many different kinds of statistical procedures and to generate data reports in various ways. Although it serves the needs of people in many disciplines and staff positions, the experiences of a large number of users suggest that SAS is not especially accommodating for beginners. Preparing information for use by SAS is akin to programming. SAS has a language of its own that is not intuitively obvious, and has strict rules for entering information.

For example, to enter variables in a SAS program, new management students have to recognize the relationship between the data on a set of sales receipts and the way in which that data must be organized for use by SAS. Because SAS does not permit spaces within variable names or variable names in excess of eight characters, students have to rename variables like order number, sales region, and product number to a format SAS can recognize: ORD∾NO, REGIONS, PROD∾NOS. (The $ symbol indicates that the values stored in the variable will consist of characters or a mix of characters and numbers rather than numbers alone.) Students must then know to precede the list of variable names with INPUT and conclude the list with a semicolon.

2. Hardware and Related Matters. Two aspects of the hardware factor emerged. The first concerns the many different kinds of terminals on campus. They vary in appearance, location of switches and control knobs, and keyboard design and labelling. Experienced
users scarcely notice the differences, but beginners are confused by them. The second aspect concerns the mainframe. Beginners are often perplexed by the complex operating system they must use in order to run a SAS program. They are also disconcerted by computer system failures or crashes, whether or not they are forewarned.

3. Diverse Instructional Efforts and Expectations. On examination, a diversity of instructional materials available to students for learning SAS in both nonclassroom and classroom settings was found. These materials generally assumed that all students had identical learning needs. However, instructors in different schools and colleges wanted students to learn SAS for various applications. Some instructors focused on statistical analysis, and others on report generation. In addition, instructors often differed in their expectations as to the level of expertise beginning users needed to achieve. Because of the diverse efforts in both classroom and nonclassroom settings, students lack common fundamental skills in the use of SAS. Therefore, instructors planning SAS-related activities at all levels face the recurring dilemma of where to begin the lessons.

4. Demands on Academic Computing Center Staff. In part, the role of the Academic Computing Center Staff, along with its student consultants, is to assist members of the university community with using the mainframe computers. The staff felt that a disproportionate amount of this service time, however, was consumed by questions dealing with elementary applications of SAS.

Discussion
On examining the above factors, it was obvious that there were some things about which nothing could be done. For example, it would be unrealistic to recommend substitution of a more accommodating mainframe program in place of SAS. Because of its overall quality and power, SAS had recently been selected as the statistical and report program to receive full Academic Computing Center support for the foreseeable future. Moreover, it was not possible to provide campus-wide standardized hardware, nor to make SAS itself or the mainframe operating system easier to use. It also was not possible to impose a common SAS instructional policy throughout the university and its schools and colleges. Finally, nothing could be done to increase Academic Computing Center staff available to help students having difficulty with SAS. It was equally clear, nonetheless, that some changes could be made by approaching the SAS problem from an instructional perspective.

At the same time, it was felt that any attempt to deal with the SAS problem should:
- benefit a large number of students,
- reduce instructors' burdens,
- withstand obsolescence,
- be both efficient and cost effective, and
- provide a common foundation for advanced SAS work.

The goal of the CID staff was to develop a means for teaching SAS that would offer maximum benefit to all interested parties-administrators, faculty, students, and Academic Computing Center staff. The staff began to consider the possibility of instruction that could be readily tailored to meet needs identified in specific courses.

Users have reported a high level of success and satisfaction with the manual. The manual reduces amount of time required to teach SAS and provides students with a common foundation for future use.

The Development Process
The instructional development process followed in the project is based on the standard CID model (Diamond, Eckman, Kelly, Holloway, Vickers, & Pascarelli, 1973). As applied in this instance, the process consisted of: (a) clarifying the problem, (b) identifying the desired SAS content, (c) determining an instructional flow, and (d) selecting an appropriate means for delivering the instruction, which in this instance turned out to be a self-instructional manual.

Preliminary Activity
Before beginning the development process, however, it was necessary to find a faculty member willing to participate in the development of an instructional program for teaching SAS. Several faculty members were teaching courses that included instruction in using SAS. One of these professors, on behalf of his department, was about to revise such a course in Management Information Systems with the assistance of CID staff. He had received an award under a Lilly Endowment Grant to the University to help support this work. Since it was clear from preliminary discussions with the faculty member that SAS instruction would continue to be a major component in the course, there was mutual agreement to make the SAS project part of the course re-design effort. The only qualification was that the new SAS design be ready in four months for use at the beginning of the next semester. Other course changes would be introduced gradually.

Course Context
Previously, Center staff had been extensively involved with departmental faculty members in redesigning the entire Management Information Systems curriculum. One outgrowth of the curriculum redesign process was the proposed modification of several courses, including the introductory Management Information Systems 255 (MIS 255) course. The course enrolls several hundred students each year, and it serves diverse audiences. In addition to being a required course for management students, it is also chosen as an elective by students from other schools and colleges in the university.

After several instructional development sessions with the instructor, the following objectives were formulated for the SAS unit:

On completion of this component, all students should be able to:
- organize given data into a form usable by the SAS program,
- enter the organized data into the computer,
- produce computer-printed reports based on this data by using appropriate SAS procedures.

Following the formulation of objectives,
The goal was to develop a means for teaching SAS that would offer maximum benefit to all interested parties.

SAS Content

The professor provided a sample of the kind of SAS-generated report he wanted students to produce. From this, it was relatively simple to identify those procedures of SAS that students in his course needed to learn. These included: Print, Sort, Link, and Merge. The next stage was to identify the sequence of steps students must follow in order to produce reports using these procedures.

Task Analysis

The correct sequence of steps was determined by working backwards from the desired final report the students were to submit. The questions were: (a) How was this particular stage of the report reached? and (b) What specific steps must be taken to reach it? Two distinct sets of activities emerged as illustrated by Figure 1.

Pre-SAS Activities

The results of the task analysis revealed that, in order to meet the needs of entry level students, the SAS instructional sequence had to begin with learning to (a) use a terminal and keyboard, and (b) interact with non-SAS aspects of the computer. These include learning to take appropriate action in the event the computer ceases to operate while in use as well as simply learning to sign on and off the computer.

The task analysis revealed another part of the sequence students must learn before actually using SAS. Few students readily see the relationship between pieces of information on a set of documents, and the need to organize these pieces systematically before entering the information into the computer. If this organizing activity, known as preparing the data step, is ignored, SAS cannot process the information as desired.

Finally, because entry errors are inevitable and revisions of data are often necessary, students need to learn basic editing techniques. Editing the information entered into the computer is, therefore, the last part of the pre-SAS sequence.

SAS Activities

The components in this sequence introduce students to those features of SAS that enable them to create the kind of reports desired by the professor. After the data is edited, students can run their first SAS jobs. Despite efforts to ensure that their input is free of errors, it is a near certainty that some students' jobs will not run properly. Typing errors, omission of critical symbols, or the incorrect choice of SAS commands are common reasons for the failure. Until students have error-free data input, they cannot obtain the required reports. It is essential therefore, that students learn what steps to take when their SAS jobs do not run properly.

The task analysis also helped reveal the specific content that would need to be taught in order for students to be able to produce the required reports. It was clear that some new concepts must be introduced. Without the combination of skills and conceptual understanding, students would lack a common foundation for advanced work in SAS, an important project objective. The next step therefore was to identify any other relevant content such as nomenclature and concepts necessary for students to be able to use SAS for Management Information Systems applications.

Associated Concepts

Identifying the procedures students must follow in order to produce reports using SAS was straightforward. Identifying the relevant concepts needed to understand the reasons underlying the procedures was more challenging. There are many concepts related to SAS. Criteria for selecting which of them to include in the instruction were also derived from the task analysis. The decision was to include only those concepts highly related to the immediate pre-SAS activities.

Concepts chosen for special emphasis included: file, permanent and temporary storage, workspace, SAS syntax, and computer crash. Crash was selected because of the unnerving effect computer crashes have on new users; SAS syntax because its correct use is necessary for correct program execution. The remaining three highly abstract concepts were chosen for special emphasis because all evidence suggested that they are: (a) persistently difficult concepts for students to comprehend and (b) essential concepts for students to understand in order to become self-reliant users of SAS.

Selection of the Delivery System

After the procedural and conceptual content were identified, attention turned to determining the most efficient and effective delivery system in relation to the following criteria:

- reduce burden on instructor,
- minimize costs,
- ensure common learning outcomes,
- facilitate learning for students.

The first three criteria, as indicated earlier, evolved from discussions with all the parties interested in the project. The fourth criteria reflects a desire to reduce barriers to student learning and to minimize sources of student frustration when learning to use SAS. Given these criteria, the previously identified content, and the specific instructional setting, a variety of instructional methods and delivery systems were considered before deciding to produce a self-instructional manual.

The Manual

The content and structure of the manual were determined by the previously discussed objectives and task analysis. The presentation of the content was heavily influenced by the work of Keller and his associates (Keller, 1983) who devised a model to assist in designing instruction that is motivating. Although description of the model is beyond the scope of this paper, its application is reflected in the manual by the use of (a) attention-getting devices, (b) relevant examples, (c) tangible evidence of success in the form of frequent printouts, (d) learner management of the pace of the instruction (e) and analogies. Expectancy for success is also fostered. Other features used in the manual are characteristic of systematically designed instruction. These include: varied presentation forms for the complex concepts; graphic representations of on-screen information; chapter
overviews and summaries; self-tests; and examples of correct printouts.

Process for Writing the Manual

Using the results of the task analysis (see Figure 1), the professor provided an initial draft of the procedural steps for the first unit. In order to check each procedure, the development staff entered the information into the computer and attempted to get the desired output. Based on this experience, the procedural steps were clarified through discussion with the instructor.

Simultaneously, the staff discussed how to incorporate the necessary concepts into the procedural steps. The scope of coverage for any one concept was deliberately limited. The intent was just to give students sufficient information so that they would have a basic understanding of what they were doing and have a sound basis for continued concept formation.

An additional concern at this same time was to develop appropriate examples to give relevance to the procedural and conceptual information. Since the majority of the students who would be using the manual were from the School of Management, examples were drawn from the business world. Attention to all these concerns was reflected in the manual in the very first draft, if only by rough notes such as "elaboration of file concept goes here." The process just described was reiterated for all six units in the manual.

Product Description

The manual is a bound seventy-nine page document (8 1/2 x 11) containing six units. All units begin with a short overview telling students what they are about to do. The bulk of each unit leads students through the steps necessary to accomplish a pre-determined outcome. Emphasis is on following correct procedures. Concepts are integrated into the procedures where appropriate, and elaborated, when necessary, at the end of the unit.

Because relevant management examples were considered important, six order forms, containing seven variables, were created (see Figure 2). These provide the data used throughout the manual. The variables are: customer number, date, salesperson, order number, sales region, product number, and quantity ordered.

Correct naming of each variable is introduced early in the instruction. The difference between string and numeric variables is discussed along with the naming conventions. Also, the distinction between an observation, in this case one order form consisting of several variables, and a variable is illustrated. Finally, students are helped to organize the information into a data matrix (see Figure 3). Students could now enter the
data into the computer. In the remainder of the manual, students learn to enter their data, revise it, and generate assorted reports through a series of exercises.

Each unit ends with a brief summary review. A crash sheet is printed on colored paper for quick reference and bound in the appendix section. A review of the logon procedure, a trouble shooting guide, short bibliography, and a summary of the computer commands used in the manual conclude the appendix section.

Formative Evaluation

When all the units were completed, five people not involved in the project and having little, if any, familiarity with SAS and the mainframe computer volunteered to participate in a one-on-one tryout of the first completed draft. Each of the five formative evaluation tryouts was conducted successively. Thus, the manual was refined during this process and all succeeding volunteers benefited from each predecessor’s experience. The final manuscript of the manual was then prepared and printed for use (Tanniru, Fiorini, & Williams, 1985). Figure 4 illustrates the flow and content of the manual.

Implementation

The manual was used initially in the 1984 spring semester. The original intent was to use it in one or two sections of the introductory course in Management Information Systems for field test purposes. Following revisions based on this field test, the manual would then be used in all sections of the course in the next semester. Once the manual was produced, however, several colleagues of the course instructor decided to use the manual in advanced courses. Because these instructors were not familiar with the manual and its intended use, a teacher’s guide was prepared for them.

Evaluation

The evaluation had a threefold purpose: (a) to test assumptions made about the computer skills of the intended users, (b) to identify any systematic differences within segments of the intended user population, and (c) to obtain feedback from both instructors and student users about the manual. Based on this information, revisions would be made to the manual in time for its use in all sections of the course for the 1984 fall semester.

Evaluation data were collected from students and instructors in both the advanced and introductory courses. Keeping in mind the intended user audience, however, particular attention was paid to the evaluation data obtained from the students and instructors in the introductory sections. The data from this smaller audience was used for assessing the user’s computer skills and individual differences among users.

Data Gathering

The data were gathered by means of an interview protocol and a questionnaire. The protocol was designed for use with all instructors employing the manual in their courses. This procedure permitted the staff to get their feedback on a uniform set of questions and to probe for additional comments on the manual. A single experienced interviewer, not previously associated with the project, met individually with each instructor to help insure consistency in data collection.

A two-part questionnaire was designed for use with the students. The first part contained 32 items. The first five items asked for demographic data. The next eight elicited the students’ prior experience, if any, with computers and with SAS. The remaining nineteen items covered: (a) the students’ experiences while using the manual, (b) perceptions of the effectiveness of the six individual

---

**Figure 2. Order form.**

---

JOURNAL OF INSTRUCTIONAL DEVELOPMENT
 units, and (c) the students' levels of confidence in their abilities to produce reports using SAS. The second part of the student questionnaire had five open-ended questions. Two questions checked the implementation and use of the manual. The other three asked each student to identify the best and worst features of the manual and to make recommendations for improving it.

Instructors were asked to have students complete the questionnaire during class time. No attempt was made to obtain responses from those absent when the questionnaire was administered, and some students did not take time to respond to questionnaires that were distributed near the end of the class period. Due to examination schedules, one class had to be omitted from the evaluation. A total of 85 students responded, 44 of whom were in the MIS 255 target group.

Results

Students. Analysis of the data generally confirmed the assumption made about the intended user population. That is, they had little prior experience in using either the mainframe computer or microcomputers for word processing, data analysis, or programming. The analysis also indicated that there were no systematic differences between: full-time or part-time students, males or females, various age groups, and those enrolled in the main campus or the university's evening school.

When asked to state the best feature of the manual, most students cited its simplicity, step-by-step guidance, and clarity of instructions. Eleven percent of the students did not respond to this open-ended question. Other favorable student comments mentioned the constant review of previous units, the helpfulness of graphics, flexible and efficient use of student time, and the examples.

On the other hand, in response to the question, "What was the worst feature of the manual?" forty-one percent of the students either wrote "none" or did not respond. Among the students who did reply to the question, there was no consensus regarding a worst feature. For example, two students mentioned the redundancy of some procedural steps as a worst feature. Four cited the cost of the manual ($6.90 for 79 pages). Two students said the manual was too helpful in that they did not have to think much about what they were doing. Several students used the space allocated for this question to suggest that the scope of the manual be expanded.

Instructors. When asked to identify the strengths of the manual, the instructors cited the simplicity of explanations, heavy provision for hands-on experience, the quality of the writing, the visualization of files, the way in which the true beginner was helped, and the general approach to the mechanics of using the SAS package. Based on the feedback from the instructors, there were no systematic weaknesses in the manual. Several, however, wished it had gone farther in scope and complexity. One said it bordered on the too-simple.

Based upon the evaluation and on consultation with the cooperating professor, no substantive changes were made in the SAS manual. Self-tests were added for each unit as well as a comprehensive final self-test to help students think about what they were doing. Also, a reproduction of the final merged SAS program was appended to Unit 6. This made it more convenient for students to check their work by looking at a single document rather than by paging through the unit for the program segments.

Conclusions and Recommendations

As of this writing, several hundred students in the School of Management have used the manual. An instructor in the School of Information Studies also chose to have students in one of her courses use it even though all examples are geared to management students. Generally, users have reported a high level of success and satisfaction with the manual. According to the instructor in charge of the management course for which the manual was created, it does reduce the amount of time required to teach SAS in that class and also provides students with a common foundation for future instruction in the use of SAS.

The findings suggest that novice computer users can effectively be taught to interact correctly with a major program on a mainframe computer through the use of a self-paced, self-teaching manual that requires minimal support from instructors. Although the content of the manual deals only with a portion of the SAS program, confidence that the instructional approach described here would be effective with all components of SAS or any other statistical program seems justified.

In order for such a manual to be effective, it is important that there be an exact match between the equipment and software used by students and between the description and operation of these items as presented in the manual. Moreover, it is essential to monitor any changes in the computing systems and revise the manual accordingly. Some computing centers frequently change
various features of their overall systems. It is equally important to monitor changes in the user population since it is quite likely that more people will be arriving at college with fairly sophisticated computing skills.

Finally, someone needs to accept responsibility for keeping the manual up-to-date by monitoring both the computing system and the student population for changes. This person needs to be identified from the beginning of the project and to become a working member of the project team. Without this level of commitment and involvement, the person is unlikely to understand the cohesive nature of the instruction and, therefore, is unlikely to be able to maintain the instructional integrity of the manual.

References

Every now and then we discover a reliable indicator or two. How do we know they're reliable? We just know. For instance, Jerry Kemp's latest book arrived the other day and I went to the shelf to get an earlier work of his on the same subject. The older book had been borrowed and not returned. Actually, it had been ripped off, which is a reliable indicator that I must improve my lending system and also that a Kemp book is a highly desirable item. Kemp's new book, The Instructional Design Process, is no exception.

The book's purpose, as its title implies, is to explain the instructional design process. Briefly, it is an introductory level textbook, pure and simple, and as such it is intended to be a curriculum guide to the Kemp method for teaching instructional design. The text is comprehensive and self-standing, i.e., can be used alone, but would likely be a better instructional tool if used in conjunction with Kemp's own audio-tutorial components (commercially available).

Kemp states in his preface that a major objective of The Instructional Design Process is "to examine a practical planning method consisting of ten elements." In Chapter Two he says that the ten elements can be illustrated by a diagram, and he promptly offers a graphic figure with eleven elements. Chapters 3 through 12 each bear the title of one of the ten elements to which Kemp referred. They are:

- Learning Needs and Instructional Goals
- Topics, Job Tasks, and General Purposes
- Learner Characteristics
- Subject Content and Task Analysis
- Learning Objectives
- Teaching Activities
- Support Services
- Evaluating Learning
- Pretesting

The eleventh element present in the diagram is an outer ring surrounding the other elements which is labeled "revision" (the ring also includes labels for formative evaluation and summative evaluation). The way that the subjects are covered explains in part Kemp's system of counting. (No, he didn't stop at ten because he ran out of fingers.) Chapter 16 is given over to "Summative Evaluation—Determining Program Outcomes." The chapter does not deal with revision as an outcome of summative evaluation. By contrast, formative evaluation is discussed for less than a page. And, sadly, there really is no meaningful discussion of the instructional designer's constant and unrelenting need to revise instruction—throughout the design phases and afterward.

The review section at each chapter's end provides questions about all of the important concepts covered by the chapter. To the author's great credit, he has attempted to always pose his chapter ending review questions on two levels: the recall level and the comprehension level. It is hard to tell how effective the questions are without trying them on a couple of classes of students, but the face evidence tends to support their validity. The answers are in an appendix at the end of the book which is useful for immediate feedback. Then, beginning with the third chapter, the review section is expanded to include application exercises as well. Many of these are related to a subject matter appendix, so that as the book progresses the student is led through the development of a training program on sky diving. Yes, sky diving. Those of you who know the venerable, white-haired Jerry Kemp will know that he is past the normal age for jumping out of airplanes, so we may assume that this choice of topic reflects his attempt to dovetail student interests and the content. Implicit in that assumption is this reviewer's own "feeling" for the text as not only being targeted for beginners, but intended for the undergraduate level. That isn't a criticism. It's an observation. We need good undergraduate texts, and properly augmented, some of them are even useful for graduate level instruction. Anyway, students in their late teens or early twenties will appreciate (and hopefully be motivated by) sky diving as the content area for their ID activities.

Every chapter begins with a set of thought provoking items. These are usually in the form of questions the typical education student or beginning teacher might ask. The tone and format of these items takes full advantage of what we know about cueing, pre-questions (in a sense), and advance organizers. The effect is to stimulate the readers' sense of inquiry rather than to send them off scanning the chapter in quest of specific, simplistic answers. Nice touch! Isn't it wonderful when one of our own skillfully applies the lessons of our field?

Other commendable traits of Dr. Kemp's text-writing formula are:

1. The frequent insertion of apt examples;
2. Good visual cueing of the text, including liberal use of headings, varied type fonts, and sundry highlighting techniques;
3. A logical organization; and
4. An index and glossary, both of which identify all of the important terms/concepts.

The book isn’t “scholarly” in the sense that every concept, theory, or statement of fact is authenticated by citing other authors and researchers. On the rare occasions where Kemp does cite another author, the reference is conveniently footnoted. At the end of most chapters is a brief listing of references, but there is no comprehensive bibliography at the back of the book. These are the features of a textbook that was systematically designed to meet (but not exceed) the needs of undergraduates. These also attributes which would be deficiencies if the book were used as a text at the graduate level.

Professor Kemp teaches common sense instructional design. As a consequence, he is a considerable distance apart from the fads and the mass of ever present but not always too practical theories that, like measles, are “going around.” His pragmatism marks him as an independent thinker. On balance, this reviewer appreciates the results of that independence, but there is one point where he’s gone too far afield. Specifically, he has established a preference for the term “instructional design” instead of “instructional development.” That’s out of the mainstream but absolutely acceptable. (Translation: the reviewer agrees.) So far, so good. But, as worrisome as the words “instructional development” may be, they deserve neglect, retirement, oblivion, or some other fate better than Kemp’s definition, quoted here: “The management of personnel, budgets, and support services to improve instruction within an organization or institution is called instructional development.” No kidding. He really said that. He even put a similar definition in the glossary.

Overall, The Instructional Design Process is better than Kemp’s earlier book, Instructional Design: A Plan for Unit and Course Development, which it replaces. An ID book with only one bad (awful, really) definition to offset a long list of pluses ought to be in every instructional designer’s library. Have you bought your copy yet? You’ll understand when I tell you I don’t feel much like lending mine.—Reviewed by Roberts A. Braden, Instructional Development, Center for Educational Media and Technology, East Texas State University, Commerce, TX.


In the introduction to Teachers and Machines, Cuban describes how his early research endeavors at Stanford University on the history of television use in the schools led him to ask questions about contemporary uses of technology by educators. In Teachers and Machines, Cuban presents his findings and describes what he perceives as education’s “fickle romance” with these technologies.

Cuban demonstrates that there are notable similarities in the way teachers and schools responded to each of these “new” technologies. In each case, these new devices were heralded by reformers as a way to revolutionize classroom instruction by increasing productivity. And in each case, the cycle of exhilaration/scientific credibility/low level implementation/disappointment was repeated.

The book contains some interesting quotations from early advocates of each of these technologies. One particularly interesting one regarding the potential impact of motion pictures on education is by Thomas Edison:

I believe that the motion picture is destined to revolutionize our educational system and that in a few years it will supplant largely, if not entirely, the use of textbooks. I should say that on the average we get about two percent efficiency out of schoolbooks as they are written today. The education of the future, as I see it, will be connected through the medium of the motion picture... where it should be possible to obtain one hundred percent efficiency.

The book contains valuable historical information on how teachers and schools reacted to these admonishments, the degree to which each new technology was implemented, and the ultimate impact of each on education and the schools. The Notes section of the book lists the material from which the information in the book was drawn, is a valuable resource.

The chapter most people will probably be interested in is the chapter on computers in schools. Here Cuban points out that although the similarities in claims, interest, and investment are “too vivid to simply brush aside as cynical... there is a danger in viewing everything as a passing fad.” After all, it is difficult to recognize a permanent and dramatic change when the metaphor of a pendulum or cycle constantly dominates our view of the public schools. Radio, television, and film have, and computers certainly will dramatically influence our lives. The question is, will computers influence the schools to the degree that they will create new patterns of living outside the schools? Cuban doesn’t really say, but he does suggest, based on his study of the past, some questions which should be used to help us answer this question. These are:

1. What is the nature of the innovation?
2. How is the innovation being introduced?
3. Who are the users and how are the machines used?

In this chapter, Cuban provides some data relevant to each of these questions. He also offers some valuable observations on several other issues related to teacher use of computers. Some of these issues are cost-effectiveness, perceived mechanisms of teaching, and impact on children. Cuban’s observations on these issues should be considered by people in the public schools as well as by people on the outside who are interested in seeing technology have a positive influence on education. Teachers and Machines will help those who see the computer as a way to improve instruction in the schools recognize the durability of classroom pedagogy and thus, be better prepared to deal with it.—Reviewed by David F. Salisbury, Center for Educational Technology, Florida State University, Tallahassee, FL.

If you are even remotely interested in corporate training or continuing education, this book is must reading. It is a thoughtful, well-documented—over 100 entries in the bibliography—review of the U.S. employer-supported education scene. It is also highly readable. No excessive footnotes (brief chapter notes are located at the end of the book) or involved periods trap the reader. The writing is clear, concise, and coherent.

An added bonus is the physical format of the book itself. Wide outer margins positively invite penciling in comments, questions, and cross-references to other materials. I also welcome the absence of high-glare glossy paper.

Ernest L. Boyer’s “Foreword” provides an excellent overview of the report. I found it refreshing to read a foreword that served as an advance organizer. Too often forewords seem to be nothing more than glorified testimonials signed by famous names.

The study itself is divided into six chapters. The first chapter provides a general overview of education and the work force. Topics include reasons for training, a general look at the differences and cooperation between corporate classrooms and higher education, lifelong learning, and the size and scope of corporate involvement in education. According to Eurih, size and scope are hard to define because there is little information available on corporate costs. Decentralization and budgeting variations make a total cost an elusive figure. The biggest cost question is whether employee wages and benefits accrued during training are counted as part of the training costs. Eurih found that these items were seldom included in analyses and evaluations of training costs. Yet, they have a dramatic impact on the final cost of training.

Chapter two is an historical overview of training in the U.S. It complements such classics as Saetler’s A History of Instructional Technology (1968).

In chapter three, the structure and methods of corporate education are scrutinized. The section on facilities succinctly portrays how the ambience of these “colleges” differs from that found in traditional colleges: older, purposeful, intense students studying on tight schedules toward explicit goals. The accommodations are also better.

The importance of training’s place in the organization hierarchy is discussed as are the instructional methods used in corporate classrooms. Instructional efficiency characterizes training. Thus, there is no one “best” method. The “best” method is the one that is most effective. Eurih also briefly outlines how courses are developed in-house. She may not have intended it as such, but I thought the section could be read as a primer by an instructional designer.

A review of the corporate commitment to research on the learning process completes the chapter. The review includes brief synopses of research results such as Kolb’s LSI (Learning-Style Inventory) and Digital Equipment’s IVIS (Interactive Video Information System).

Chapter four centers on the curriculum and its quality. Eurih identifies five major curriculum areas in corporate education: basic skills, management/executive training, technical/scientific study, sales/service/customer training, and general education. The management/executive curriculum itself covers four general areas: managing time, managing people, managing money, and managing production and operations. Each of these areas is discussed with implications for the K-12 curriculum. Current MBA programs also receive some suggestions. The corporate emphasis on course evaluation forms the basis of Eurih’s discussion of course quality.

Chapter five shifts the focus to corporate colleges granting academic degrees. The colleges along with such information as sponsor, history, status (e.g., independent, nonprofit), date established, accreditation, and degree awarded are listed in a table. Eurih examines four of these colleges—American College, Wharton Institute of Graduate Studies, Rand Graduate Institute, and The American Institute of Banking (at Boston)—from the perspective of “Why start a degree program?”

She identifies five general characteristics of corporate colleges. Most are nonprofit. They offer a range of degrees. They have “solid credentials.” i.e., they are accredited. Their administrative terminology (e.g., deans, faculty, catalogs, graduation requirements) parallels that of traditional institutions. And, they continuously evaluate faculty and courses.

The last chapter deals with emerging issues in traditional education, the learning environment, and the need for a comprehensive view of all our educational resources. Eurih suggests the formation of a Strategic Council for Educational Development to help guide educational improvement and national policies for the next decade.

Of special interest to researchers in the field—and doctoral candidates in search of a dissertation topic—are the areas Corporate Classrooms identifies for further study. For example, in discussing training’s emphasis on team work, Eurih raises questions such as “. . . have schools and colleges been so individually oriented that students have not learned to work with others?” (p. 65) She submits that “a study of proprietary schools and trends is long overdue” (p. 118).

Regardless of the aspect of corporate education under discussion, Eurih never loses sight of the broader picture. Constant comparisons between corporate and traditional education highlight the advantages, disadvantages, and implications of and for these two major sources of education in our society.

Corporate Classrooms is timely and sheds light on issues that are often overshadowed by the “system of magic-value ratings of various subjects” (Pediwell, 1939, p. 79). Given the current debate on the mission of higher education apropos career-oriented versus liberal arts education, you cannot afford to miss this book (and at $8.50, you can definitely afford it)—Reviewed by Esther R. Sinofsky, Instructional Design Consultant, Los Angeles, CA.

References

Pediwell, J.A. (1939). The saber-tooth cur-

1986. VOL. 9. NO. 1

So what does “top management strategy” have to do with the instructional designer, you ask? Indeed, I am sure many who are not top managers are turned off by the title, which is a shame. Once in a while I come across a book that drastically changes my thinking. In this case, the serendipity occurred while I was browsing through the business section of a local bookstore. I picked up this book and couldn’t put it down! Why this strong reaction? Because the basic thesis of this book is that managers spend too much time on operational planning and not enough on strategic planning. Tregoe and Zimmerman think this accounts for a lot of business failure in American today. I would like to add that this same tendency has resulted in much of the failures that have occurred in instructional technology. That is why this book is so important.

Strategy, explains Tregoe and Zimmerman, establishes the framework that defines the organization and provides the basis for its key choices. Without an adequate strategy an organization is bound to go from decision to decision without consistency. Strategy provides that consistency. Most organizations are so involved in their daily (weekly, monthly, and annual) operations that they fail to develop a strategy (or ignore the strategy that had been formulated long ago). They may perform well operationally, but they lack an overall direction, which eventually will come back to haunt them. I can think of organizations such as Atari, Commodore, and many of the airlines that boomed in the years of prosperity because of good ideas, good timing, and good operational planning. However, when demand slackened off or preferences changed, they could not adapt. They lacked the carefully forged strategic plans that have made the IBMs and other “excellent” companies so successful.

So, you are still asking what this has to do with you. Well, how many of us think about the difference between strategy and operations? How many of us have embraced a new technology without ever thinking about whether it fits into the “raison d’etre” of our organization or our own needs? Every technology that I can think of that has been disappointing in its implementation (and that’s every one) has been disappointing because we lacked a vision for how it should be used. We had lots of visions, but most of them were dreams dreamed by starry-eyed zealots of some exciting technology or another, not the systematic visions of a strategic planner who knows precisely what is needed and why.

An effective strategy must give direction and deal with not only what the organization is, but what it wants to be. A strategy must also be communicated throughout the organization, and others must share the same vision. Strategy cannot be kept in a drawer; it must be on top of everybody’s desk. It should guide all major decisions, short-term as well as long-term. The strategy should be based on sound assumptions about the organization and its environment. And it must be reviewed and revised regularly on the basis of new information when it becomes available.

Educational organizations tend to be weakest strategically. I can think of so many examples of how technological innovations were misused because of a lack of strategic planning and/or understanding. Every organization is going to differ in strategic orientation. Unless we, as instructional technologists, understand organization strategy and use it in our own planning, we are destined to fail in our efforts at change agency.

Top Management Strategy was written by two men who are qualified to write about the subject. Tregoe, with Charles Kepner, is the co-author of the best problem-solving course available and one of the most important books on management, The New Rational Manager.

You will find this book has a lot to say to the instructional technologist who is not even near top management levels, because we are often dealing with organizations whose top management has neglected to establish, communicate, or use strategic planning effectively. We should be aware of the ramifications of this. Our survival (and mental health) depends on it.—Reviewed by Dr. Dean R. Spitzer, President, High Impact Training, Alameda, CA.
A study was conducted to examine the procedures and requirements of a formative evaluation when it is carried out for instructional materials that have been designed for a U.S. target audience, but which also have potential for use by a population of another country such as Brazil. The study concentrated on determining whether different formative evaluation procedures should be implemented, and, if so, identifying possible additional procedures to be carried out during evaluation. Two sample populations were used in the study: 49 U.S. learners and 49 Brazilian learners who were professionals in similar areas of expertise and training. The two groups were given an instructional module on problem solving which was produced in two languages, Portuguese and English. Their performance on the modules, which contained a pretest, a posttest, and an attitude questionnaire, was compared under three conditions: one-to-one, small group, and field trial evaluations. The two main dependent variables assessed were the characteristics of the feedback and the revision decisions. Findings indicated that: (a) the same evaluation procedures used to collect data to revise instructional materials designed for audiences that differ in cultural traits or values can be applied to that effect, (b) a carefully performed one-to-one evaluation provides substantial data for modification of the instructional materials, so that small group evaluation may not be required. In addition, two extra procedures might prove useful: (a) after initial translation to the required language, a second translation back to the original language to identify any improperly translated words, and (b) material revision by evaluators prior to implementation to discover any cultural inadequacies. A list of references is appended. —Microfiche 75 cents, paper copy 1.80 plus shipping, as document ED 263 908.


For this discussion, new technologies are defined as specially designed training systems based on microcomputers that incorporate high resolution color displays, special input devices for responses, laser videodiscs for storage of stimulus materials, and hard disk storage for programs and responses. Such systems have several advantages over existing training options in business and industry—e.g., reduced time needed for training—and more effective, individualized instruction as a result of utilizing computer based training. In addition, the computer can track, analyze, and present results quickly and meaningfully. Computer simulations provide the opportunity to present trainees with experiences that go beyond those available through textbook materials and classroom setting. However, three major issues should be considered when introducing new technologies into industrial training programs: (a) the needs of those who will be using the system, (b) the attitudes and reactions of trainees, and (c) the attitudes and reactions of the trainer. A list of references is provided.—Microfiche 75 cents, paper copy $1.80 plus shipping, as document ED 263 900.


Two distance education courses presented via teleconferencing by the University of Calgary were redesigned to include graphic support utilizing Canada’s Telidon videotex system. Data were gathered on both the instructional effectiveness of, and student responses to, the graphics. While the evaluation identified a number of features that needed to be modified in the system, it also revealed aspects of a computer graphic that make it effective. This report provides a brief overview of instructional graphics, describes the Calgary experience with Telidon page design, and looks at ways in which instructional graphics can be used to control various aspects of instruction. Conclusions indicate that the Telidon system lends itself well to the presentation of simplified graphic materials for illustrations, cueing, and structures, and that the use of such graphics to control the flow of instructional activities is very effective. Although limitations imposed by text size when the monitor is used with groups of students would make more complex and detailed graphics illegible at the present time, future developments leading to more sophisticated computer graphics systems will allow more complex animation and screen resolution in the future. A list of references is included.—Microfiche 75 cents, paper copy $1.80 plus shipping, as document ED 263 883.

The above documents may be ordered from the ERIC Document Reproduction Service (EDRS), 3900 Wheeler Ave., Alexandria, VA 22304-5110. Please order by ED number, indicate the format desired (microfiche or paper copy), and include payment for the price listed plus shipping. For information on shipping charges, call EDRS at 1-800-227-3742. Inquiries about ERIC may be addressed to the ERIC Clearinghouse on Information Resources, School of Education, Syracuse University, Syracuse, NY 13244-2340.

JOURNAL OF INSTRUCTIONAL DEVELOPMENT