This is to certify that the

dissertation entitled

CASE STUDY OF THE INTEGRATION OF TECHNOLOGY INTO AN EXISTING COURSE BY AN INSIDE DEVELOPER

presented by

JAMES OLIVER NEWTOWN

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in CEPSE

Stephen F. Gelon
Major professor

Date Oct 16, 2000

CASE STUDY OF THE INTEGRATION OF TECHNOLOGY INTO AN EXISTING COURSE BY AN INSIDE DEVELOPER

By

James Oliver Newtown

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSPHY

Counseling, Educational Psychology and Special Education

2000

ABSTRACT

CASE STUDY OF THE INTEGRATION OF

TECHNOLOGY INTO AN EXISTING COURSE

BY AN INSIDE DEVELOPER

By

James Oliver Newtown

There is a continued interest in using computer technology in education. However, most of the emphasis has focused on an "outside" developer, consultant or team that will develop and implement a complete new course or lesson that will use this technology. Little attention has been paid to the potential "inside" faculty developer.

The purpose of this study was to examine the development of a computer instructional aid by an "inside" faculty developer and to compare the process with the models of outside developers. The study focuses on the development and diffusion of a simple drill-and-practice program created for a multi-section introductory computer course at a post-secondary educational institution by an instructor of that course. This was a 20-hour course concentrating on basic computer terminology, concepts and processes with a standard "common" final exam.

The faculty developer wrote a drill-and-practice computer program as a computer instructional aid to assist students in preparing for the final exam. Three instructors, including the developer, piloted the distribution and use of the computer instructional aid. The aid was then distributed to 136 students in six sections of the course taught by four instructors. Interviews were conducted with instructors before and after the course to examine instructor attitudes regarding the aid. Students were asked to complete a

questionnaire at the end of the course, after they had taken the final exam to find out if they found the aid useful. Final exam grades were collected to identify possible effects of use.

Findings included the importance of the size of the project and the inside developer's knowledge and experience. The inside developer must be very cautious in evaluating the amount of time and effort required for the project and whether it is justified. Not only must the inside developer be knowledgeable regarding the course and the organization, but also the areas of instructional design, technology and diffusion of innovation. The final conclusion was that the inside developer can create a useful educational tool but the developer's knowledge and the size of the project must be carefully evaluated.

ACKNOWLEDGEMENTS

I sincerely wish to thank everyone who assisted me with this project. I am particularly grateful to Dr. Stephen Yelon who never gave up on me. I also wish to thank the other members of my dissertation committee who provided guidance, inspiration and valuable criticism: Dr. Chris Reznick, Dr. Ralph Putnam, and Dr. Patrick Dickson.

Very special thanks to the students and instructors who participated in this project including Lori Johnson, Joyce Newtown, Judy Robbin, Tina Nies, and Cynthia Townsend. Of course, I also wish to thank my many friends and relatives who inspired and encouraged me in this effort particularly my parents, Ralph and Jean Newtown.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

The purpose of this study is to examine the development and diffusion of a computer instructional aid for an existing multi-section college course by an "internal" change agent and to compare that process with the processes of instructional design, educational technology, systems development life cycle (SDLC) and diffusion of change.

The Call for Using Computer Technology in Education

Administrators and technology advocates at all levels of education have accepted computer technology and are looking for new ways to use these tools. Noble (1998) states that "the high-tech transformation of higher education is being initiated and implemented from the top down" (p. 23). Kearsley (1998) wrote that "technology has become the great siren song of education" (p. 47). Unfortunately, administrators often require the use of these computer tools without an understanding of the process involved. This includes the lack of "any student and faculty involvement in the decision-making" (Noble, 1998, p. 23), "little background and fewer resources with which to conduct ... planning" (Lawler, Rossett & Hoffman, 1998, p. 29), or an understanding that "educational technology is neither inherently beneficial nor excessively costly to organizations" (Leh, Sleezer & Anserson, 1998, p. 32). Often instructors and instructional designers are expected to replace an entire live course with an alternative that relies heavily, if not completely, on computer technology. Cavalier and Klein (1998) noted that "computer-based instruction is becoming a widely accepted instructional delivery

medium in schools" (p. 5). Firdyiwek (1999) discusses the use of computer "tools that purport to handle the design, delivery, and management of whole courses online" (p. 29).

There Are Many Ways To Apply Computer Technology To Education

Educators often use computer technology to replace typical classroom instruction (Tannenbaum, 1999; Scanlon et al, 1998; Kearsley, 1998). Sometimes instructors use computers as one of many media to supplement material presented by the instructor, much as a textbook is used." Computer Assisted Instruction or "CAI has been the most researched of the various interactive technologies. It has been demonstrated to be an effective supplement to traditional classroom instruction" (OTA, 1988, p. 42). According to Hentrel and Harper (1985), "Computer-assisted instruction used as a supplement to regular classroom instruction has been found to be consistently effective" (p. 69). These supplementary computer applications may not seem as elaborate or as complex as the most recent computer games and simulations but they are useful. For example, a simple drill and practice program may benefit students and be a satisfactory choice for aiding practice.

Who Provides the Computer Based Instructional Aids?

Complete computer tutorials often require extensive development. Estimates range up to 100 or even 300 hours of development time for one hour of student use (Tannebaum, 1999). As a result, applications must usually be purchased from software companies that can afford specialists who will invest large amounts of time in developing these applications. However, knowledgeable instructors may be able to afford the

minimal time necessary to create simple instructional aids. Instructors are in the unique position of already being intimately knowledgeable about the course content, the students and the goals of the course compared with external instructional designers and computer specialists. If the application is for a multi-section course also taught by other instructors, the instructor/developer also has many advantages as an internal change agent compared with an external consultant or developer (Altimirano, 1998; Havelock, 1995) Thus if an instructor sees an instructional need in a multi-section course that can be satisfied with a simple computer program, he or she may be more likely to develop an application that will be easily accepted by their colleagues and students.

Why Isn't Technology Used More In College Education?

There may be a problem in introducing technology into an existing course. Relatively autonomous teachers in existing successful, or at least uncriticized courses may resist the integration of new technology if they see it as a threat to the integrity of their course, if they see their control and authority challenged. On the other hand colleagues may be more willing to accept something made by a peer. Unfortunately, most teachers do not have the time or skill to make specific tailored computer based instructional tools and aids.

Even for teachers with some computer programming skills, most would not know where to start and what to do. Almost all process models for development and diffusion assume an external developer and change agent (Wilson, 1999). The differences between an inside developer and an outside developer are not specified. The purpose of this study is to specify the "insider" approach.

Why Don't More Teachers Create Their Own Computer Based Instructional Aids?

Creating computer instructional aids does require some minor computer skills and programming skills. The computer skills include things such as being comfortable working with a computer, the ability to start and run programs, the ability to copy and move files, and the ability to be able to accurately follow specific written directions. If a basic modifiable program is available, required programming skills are actually minimal: (1) the ability to load, edit and save files, much like in a word processing program; and (2) a general understanding of the compiling process which converts a text file into an executable program.

But believing they can create computer instructional aids requires more than just the necessary skills. This is where anecdotal evidence from colleagues and a detailed descriptive process can be important in creating the right attitudes and environment for diffusion of innovation. The support of statistically significant differences in outcome scores is probably of only minor importance to teachers and students compared with perceived usefulness.

Once belief has been created, an upward supportive cycle can be created. Instructors are aware of what they know and can do; see an opportunity to apply these skills; use these skills to create even a simple computer instructional aid; and are successful. In each cycle an instructor gains experience in using these skills and applying them to a situation. As a result, the instructor also gains confidence and enthusiasm.

Time limitations are probably the biggest obstacles instructors must face. In order to justify an investment of time, the results must either be dramatic or the amount of time

invested must be minimal. As noted before, typical sophisticated CAI has development time to student usage time ratios of 100 to 1 up to as high as 300 to 1. This sort of effort can be justified only if hundreds or even thousands of students can use the CAI, which is usually the case only if the CAI is of a very general nature, such as in a course that all students in a school, district or college must take. But CAI doesn't have to be sophisticated to be useful.

Researchers seem unable to consistently find statistically significant differences in outcomes between students using CAI and those not using CAI. "When CAI is used as the sole basis for instruction, the results are mixed" (OTA, 1988, p. 46). Because of the complex process of learning and each individual mental construction, it is difficult to demonstrate the precise learning contribution of a computer instructional aid. Does that mean CAI may not be very useful? Not if researchers can show usefulness of the "aid."

We can say that a component is useful if the students and instructor perceive it as being useful. This perception can be measured by student and instructor responses to questionnaires, students willingness to invest time is using the aid, and instructor willingness to invest time in developing and introducing the aid to students. The aid is useful if it is perceived as being useful and is used. Since we are dealing with adult students, who better to decide if it is useful or not?

When a tool is useful, a learner consistently and consciously decides to employ the tool to meet a goal and believes the tool is helping to make progress toward the goal. Given that a computer instructional aid fulfills a carefully designed instructional function, such as objective oriented practice, the goal of the insider diffusion agent is to promote the perception of usefulness in the eyes of instructors and students.

Range Of Computer Instructional Aids

The range of computer instructional aids extends from simple drill-and-practice programs to extensive multi-media interactive tutorials and simulations (Resta, 1984; OTA, 1988; Gayesli, 1999). Educators tend to focus on the more elaborate and attractive aids that use the latest technology. For a long time it has been known that the more student senses involved in the learning experience, the more students pay attention and the more they remember (Fleming, 1987). As the old saying goes, "we retain: 10 percent of what we read; 20 percent of what we hear; 30 percent of what we see; 50 percent of what we hear and see; 70 percent of what we say; 90 percent of what we say and do" (Anonymous, 2000). Therefore, educators and students are more attracted to elaborate and attractive aids. However, simpler appropriate aids can still be useful, sometimes just as useful, as sophisticated "glitzy" CAI. In addition, simpler aids require a smaller investment in development time.

Research Purpose

The purpose of this study is to describe how an "inside" instructor developer creates, develops and integrates a relatively simple technological change into an existing multi-section course so that it is seen as useful by other teachers and students. In addition, the researcher will abstract the process and compare it to present methods of development and diffusion as done by outside agents.

CHAPTER TWO

REVIEW OF LITERATURE

Purpose of the Review

This study examines the process of development and diffusion of a computer instructional aid by an inside developer. As a result this study touches on many areas. These include instructional design, educational technology, development of computer systems including the Systems Development Life Cycle (SDLC) and diffusion of innovation. This chapter will focus on the major foundations for this study in these areas, emphasizing typical process descriptions so they can be compared later to the description of this study.

Instructional Design

One of the primary concerns of instructional design has been with identifying and analyzing the type of knowledge that is being taught. Bloom's taxonomy (1956) started by dividing educational objectives into cognitive, affective and manipulative or psychomotor. He then further subdivided cognitive objectives into six major classes or levels:

> "1.00 Knowledge
> 2.00 Comprehension
> 3.00 Application
> 4.00 Analysis
> 5.00 Synthesis
> 6.00 Evaluation" (p. 18).

Similarly, Gagne identified categories of learning. "These are as follows: (1) verbal information; (2) intellectual skills, having five subordinate types called discrimination,

concrete concept, defined concept, rule, and problem solving; (3) cognitive strategies; (4) motor skills; and (5) attitudes" (Gagne, Wager, & Rojas, 1984). Gagne and others reference these categories in many other writings (Aronson & Briggs, 1983). Specifically, Aronson & Briggs (1983) state, "the capability that one acquires when learning *verbal* information (e.g., a spouse's birthday) is stating the information" (p. 81). This is the type of subject matter or content that I used in the test bank portion of the computer instructional aid created in this study. The test bank primarily consisted of terms and definitions. This is content that falls into Bloom's knowledge category and Gagne's verbal information category.

The field of instructional design has also been concerned with identifying the key elements of a good lesson. The key example of this is Gagne's events of instruction (Gagne et al., 1984):

"1. Gaining attention
2. Informing learner of lesson objective
3. Stimulating recall of prior learning
4. Presenting stimuli with distinctive features
5. Guiding learning
6. Eliciting performance
7. Providing informative feedback
8. Assessing performance
9. Enhancing retention and learning transfer" (p. 59).

Practice, the purpose of the computer instructional aid created in this study, is incorporated in event 6, eliciting performance, and event 7, providing informative feedback. These events are also listed in several other publications (Aronson & Briggs, 1983; Gagne, 1977; Gagne, Briggs & Wager, 1992) and have become a fundamental element of instructional design. Gagne et al. (1984) also related Gagne's learning

outcomes with his events of instruction to provide guidelines for creating complete CAI (Computer Assisted Instruction) lessons.

Instructional design theorists have also tried to identify the procedures and process needed to create good instruction. Rosenberg et al. (1999) describe the generalized systematic model for Instructional Systems Design.

> First, the instructional requirements of both the learner and the task or job are analyzed, to determine the precise instructional need. Next, an instructional program is designed, with objectives and testing that are linked to the preceding analysis. Instructional materials are then produced and delivered according to the design. In each phase, evaluation data are collected and revisions are made so that the outcome of the process meets the identified need as closely as possible. One such systematic model has come to be known simply as ADDIE: analysis, design, development, implementation, and evaluation. Today many operational ISD models exist in practice, but most either can trace their roots to ADDIE or accept the ADDIE concept as a foundation. (pp. 27-28)

Davis, Alexander and Yelon (1974) used a Gantt chart to show the sequence and parallel nature of individual steps that were summarized under the general headings of analysis, design and evaluation. Another early example was Kaufman's (1972) attempt to apply a systems approach to the instructional design process. He listed six specific steps:

1. Identify problem (based upon need)
2. Determine solution requirements and solution alternatives
3. Select solution strategy from alternatives
4. Implement select strategy
5. Determine performance effectiveness
6. Re-do as required

Upon close examination, you can see that the first five steps do correspond to the ADDIE stages.

The ADDIE model provides the basis for many other models used to describe the process of designing instruction. Dick and Carey (1996) identified a ten-step procedure for designing instruction in The Systematic Design of Instruction. Although not intended

to be strictly linear, the steps are numbered here for reference purposes. Dick and Carey's model consists of

1. Determine instructional goal
2. Analyze the instructional goal
3. Analyze learners and contexts
4. Write performance objectives
5. Develop assessment instruments
6. Develop instructional strategy
7. Develop and select instruction
8. Design and conduct the formative evaluation of instruction
9. Revise instruction
10. Conduct summative evaluation (pp. 5-7)

Dick and Carey (1996) describe the above steps as "a systems approach for the design, development, implementation, and evaluation of instruction" (p. 4). They also tend to focus on instructional materials, especially "learning activity packages (LAPs) and modules" although they just use the term instruction. They define this instruction or module as "a self-instructional printed unit of instruction that has an integrated theme, provides students with information needed to acquire and assess specified knowledge and skills, and serves as one component of a total curriculum." This level of instruction or module could be a lesson or an entire course. However, it would typically not be one part of a lesson such as the computer instructional aid developed in this study to assist with student practice. While Dick and Carey try to describe the general development of instruction, they acknowledge that their model is "most directly applicable to the development of print instruction" and that often it is applied to instruction that includes a computer.

Dick and Carey also advocate the use of their systems approach model by both an individual instructor and an instructional designer who "often works with a team of specialists to develop the instruction" (p. 11). While they identify these two groups of

potential users of their model, they make no distinctions between the individual instructor, who I refer to as the inside developer, and the instructional designer, who I refer to as the outside developer.

In their detailed description of the step to "develop and select instruction," Dick and Carey explain that their concern is to "develop and select instructional materials" (p. 225). While there may be existing materials that can be used, they provide a detailed list eleven specific sub-steps that can be used to develop instructional materials:

1. Review the instructional strategy for each objective in each lesson.
2. Survey the literature and ask subject-matter experts to determine what instructional materials are already available.
3. Consider how you might adapt available materials.
4. Determine whether new materials need to be designed. If so, proceed to step 5. If not, begin organizing and adapting available materials, using the instructional strategy as a guide.
5. For each lesson, consider the best medium to present the materials, to monitor practice and feedback, to evaluate, and to guide students to the next instructional activity, whether enrichment, remediation, or the next activity in the sequence.
6. Determine the format and presentation procedures for each objective or cluster of objectives. Plan any general format or presentation pattern you believe is necessary or would be effective. Plan the actual script and illustrations for the instructional strategy.
7. Write the instructional materials based on the instructional strategy in rough form. You will be amazed at how stick figures and rough illustrations can bring your ideas to life or a first trial. Printed, visual, or auditory materials in this rough form will allow you to check your sequence, flow of ideas, accuracy of illustrations of ideas, completeness, pace, and so on. Make a rough set of materials as complete as possible for each instructional activity.
8. Consider each completed lesson or class session for clarity and flow of ideas.
9. Using one complete instructional unit (complete instructional program), write the accompanying instructions to the students for that activity if they are required.
10. Using the materials developed in this first inexpensive, rough draft, you are ready to begin evaluation activities. Chapter 10 introduces and discusses procedures and activities for evaluation and revising instructional materials.
11. You may either develop materials for the instructor's manual as you go along or you can take notes as you develop and revise the instructional presentations and activities. Using the notes, you can later write the instructor's guide. (pp. 230-231)

Note that step 2 above refers to consulting with "subject-matter experts." If the inside developer is an instructor in the area involved, as I was, he or she is probably already a subject-matter expert. Also note that steps 8 and 9 specifically indicate the development of a complete unit, not the development of one component within an existing course, which is the focus of this research.

Some theorists have created models that do not relate to the ADDIE as easily. Some of these models tended to summarize the instructional design process. Keller (1987) created a simplified consensus model for the process of instructional design. He states that the instructional designer "begins with a statement of broad goals, moves on to an analysis of who will be taught, performs a task analysis, develops and presents materials, evaluates the effectiveness of the teaching" (p. 16).

Gagne et al. (1992) adopted Dick and Carey's overall ten-step procedure and summarized it by stating that "all the stages in any instructional systems model can be categorized into one of three functions: (1) identifying the outcomes of the instruction, (2) developing the instruction, and (3) evaluating the effectiveness of the instruction" (p. 21).

Brethower and Smalley (1998) focusing on the design phase, expanded it but combined several other phases and excluded development completely. They ended up with the four phase model (p. 149):

1. Business-Needs Specification
2. Performance-Requirement Specification
3. Design Phase One: Specification of Work Processes
4. Design Phase Two: Specification of Instructional Processes
5. Implementation and Evaluation (p. 149)

Other theorists have expanded on the basic ADDIE model producing detailed steps similar to those of Dick and Carey. Gustafson and Tillman (1991) listed ten steps, many of which match those of Dick and Carey:

1. Identify needs and goals
2. Organize the course
3. Write objectives
4. Prepare assessments of learner performance
5. Analyze objectives
6. Design strategy of instruction
7. Design lessons or modules
8. Conduct formative evaluation
9. Install course
10. Conduct summative evaluation (p. 10)

Again, these steps are numbered here for reference purposes. Gustafson and Tillman specifically stated, "these ten steps are usually *not* performed in lock step fashion" (p. 10). Note that several of the steps, specifically 2, 7 and 9, clearly indicate that his process relates to the development of a complete course or module.

Mager (1997) reiterated the ADDIE model, although he did combine two of the phases: "The procedures through which instructional design and development are carried out are often clumped into four broad phases: analysis, design/development, implementation, and evaluation/improvement" (p. 13). Mager also directly addressed the fact that these models primarily address a complete learning module not just a part.

> The procedures described in this book appear in the approximate order followed when a *complete* instructional development project is undertaken, that is, when all the procedures are used. But it is *not necessary to apply them in the exact order shown*, nor is it always necessary to use *all* of them to accomplish your mission. (p. 16)

Clearly there are many system models. Most of them have a great deal in common and owe much to the ADDIE model, although there are some differences. Gustafson (1991) summarized five models including the Instructional Development Institute (IDI)

model (Twelker et al., 1972), the Interservices Procedures for Instructional Systems Development (IPISD) model (Branson, 1975), a version of the Dick and Carey model (1990), the Seels and Glasgow model (1990), and the Robert Diamond model (1989). I have combined these models in Table 2.1. Gustafson emphasizes that these systems development models or "instructional systems models are characterized by four key features: (1) large scale team development, (2) a linear develoment process, (3) wide distribution of the system, and (4) a problem solving orientation" (p. 31).

Gustafson (1991) also summarized three product development models: the Van Patten model; the Leshin, Pollock, and Reigeluth model; and the Bergman and Moore model. I have summarized these in Table 2.2. These models can be compared with the systems development models in Table 2.1, however their focus is more on creating a final product than the process. Gustafson states "product development models are characterized by four key features: (1) they usually assume that the instructional product is needed; (2) they assume something should be produced rather than selected or modified from existing materials; (3) they place considerable emphasis on tryout and revision; and (4) they assume that the product must be usable by a variety of 'managers' of instruction" (p. 22).

But Instructional Design models "rely on outside expertise" (Wilson, 1999) and focus on development of a complete instructional module. In this case only a small element of instruction is being created. This means that most of the other elements will not be changed and increases the importance of consistency in instructional design. The new component, in this case the computer instructional aid used for practice, must be consistent with the major components: the objectives, the content, the methods (which

includes practice along with other components), the evaluations and, of course, the desired real-world performance (Yelon, 1996).

Table 2.1
Systems Development Models from Gustafson

IDI Model	IPISD Model	Dick and Carey Model	Seels and Glasgow Model	Diamond Model
Identify the problem		Identify instructional goals		
Analyze the setting	Analyze	Conduct instructional analysis	Problem analysis	
Organize management		Identify entry behaviors and characteristics	Task and instructional analysis	
Identify objectives		Write performance objectives	Objectives and tests	Determine objectives
Specify methods	Design	Develop an instructional strategy	Instructional strategy	Select instructional formats
			Media decisions	Evaluate and select existing materials
Construct a prototype	Development	Develop and select instructional materials	Materials development	Produce and field test new and available materials
Test the prototype		Design and conduct formative evaluation	Formative evaluation	
Analyze results				Coordinate logistics for implementation
Recycle or implement	Implement		Implementation maintenance	Implement, evaluate and revise
		Design and conduct summative evaluation	Summative evaluation	
	Control	Review	Dissemination diffusion	

Table 2.2
Production Development Models from Gustafson

Van Patten Model	Leshin, Pollock, and Reigeluth Model	Bergman and Moore Model
Analysis	Analyzing needs	Analysis
Design	Selecting & sequencing content	Design
Development	Developing lessons	Develop
Pilot test	Evaluating the instruction	
Revision		
Production		Production
		Author
Duplication		
Implementation		Validate
Maintenance		

Education and Technology

Education has a history of becoming fascinated with new technology. Each new technology brings a wave of excitement, often with the hope that this technology will be the miracle one that will solve all of our problems. Educational technology writers have often focused on the effect and impact of a new technology on society and specific institutions such as schools (e.g. Perelman, 1987; Provenzo, 1986). Many of the early researchers in the field of educational technology focused on the unique factors of different technologies and how they affected the educational institutions and processes (e.g. Nugent, 1987; Weizenbaum, 1976).

However, there have been few attempts to identify a detailed procedure by which technology can be incorporated into an educational process. This absence may be partly due to the fact that the field of educational technology is relatively young. In the early days of any field of study the focus is on specific unusual factors and their affect. In addition, individuals and small groups working on small simple projects have done much of the work in educational technology. However, the need for a formal procedure becomes more obvious when large groups are working on complex projects such as the development of complete lessons or modules.

Rizza (1981) examined a "systems approach to courseware development" (p. 278) which used the five stages of the ADDIE model (analysis, design, development, implementation and evaluation). He described how each step of this model could be followed to develop computer-based educational materials. His approach clearly focused on the development of a complete lesson by an outside expert. He said, "to assume that a teacher can develop a high quality computer-based education offering in his or her spare

time is equivalent to asking teachers to write their own textbooks, script and record their own multimedia materials, stage and deliver their own television shows, and develop and analyze their own standardized tests" (p.276). Note that the key restriction in this argument is the amount of time required.

Keller (1987) described an eight-step process model for the design of courseware. He defines courseware as "the teaching materials with which students interact when they're taught by computers" (p. v). He elaborates by saying that "courseware consists of everything that goes into the creation of a tutorial lesson delivered on a computer, from deciding what is to be taught to determining the best way of presenting it" (p. v). His eight steps are:

1. see the problem in system terms
2. specify a solution
3. code and make preliminary tests
4. fine tune for easier use
5. do a formative evaluation
6. revise as needed
7. implement
8. perform a summative evaluation and follow-up (p. 16)

Keller bases this courseware design model on the more general instructional design models of Gagne and Briggs (1974); Kemp (1977); Dick and Carey (1978); Romiszowski (1981); and Rowntree (1982). Keller does not specify that this process is intended only for outside expert consultants but "for people who are interested in producing their own courseware" (p. v). He further states "readers need not be computer experts to write the kind of courseware that appears in the first part of the book. They should, however, be familiar with at least the basics of programming" (p. v). However, his definition of courseware does indicate his process relates to the creation of a complete lesson or "unit of courseware" (p. v).

There are also problems with evaluating the effectiveness of using computers in education. As early as 1974, Jamison et al. summarized several studies by concluding that the best conservative statement that could be made about the effectiveness of Computer Assisted Instruction is that it is at least as good as traditional instruction. Not a ringing endorsement. Clark (1985) concluded that "with studies where only one faculty member designs both the computer and the conventional treatment are considered, the learning advantage for computers reduces to insignificant levels" (p. 146). Clark is not opposed to the use of computers in education however, he does warn of the dangers of expecting "sizeable learning and performance gains" (p. 146).

> Of course the problem of confounding or construct validity does not imply that computers are useless in education. To the contrary, intelligent and highly practical applications of computers to education are possible. Computer enthusiasm is a source of support for curriculum reform, student and community motivation and individualized instruction. However, researchers and educators should be reluctant to attribute learning gains either to the computer or to any unique quality of computers. (pp. 146-147)

Educators often perform much of their work with no connection to their peers within and without the educational environment. While business workers must continually be aware of their role in the overall scheme of things, educators are often isolated in their classroom and in a specific area of knowledge (science, math, etc.). This isolation does not encourage the adoption of an overview of the entire process such as that contained in the systems development life cycle (SDLC) often used in business. There are substantial differences between the prime motivations of educators and business computer system developers. While business is concerned with making a profit, education is concerned with learning. However, educators can benefit by examining and comparing what is done in education with what is done in the business world.

The major concern of the business world has always been to maximize profit by increasing benefits, such as revenue, and to decrease costs. Applying this concern to the development of computer systems and software has meant that business has always actively sought ways to become more efficient. This included the introduction of the principles of structured programming and structured design in the early 60's. This also included using a systematic approach to developing computer systems.

Frederic Taylor (1912) attempted to apply scientific principles to management. This general process has been adopted and modified by many writers concerned with making the development of computer systems more efficient. From this effort a Systems Development Life Cycle (SDLC) has been identified for describing the steps necessary to develop a computer system. Although each writer's list of steps is a little different, the similarity can be seen in Table 2.3. Note that these lists of steps attempt to define a step-by-step procedure for developing computer systems. This is an important distinction for even though writers will examine and study a process, they sometimes do not create a step-by-step procedure to guide others in working through that process.

Developers of business computer systems are probably concerned with creating a step-by-step procedure because of the complexity of their task and the need to manage a large process. Business computer systems are often large and affect many people and organizational departments. It is often easy to get so concerned with the minutiae of a project that one loses an overview of what needs to be done to complete the project. An overall step-by-step strategy can help guide one through the morass of detail. A detailed procedure can also help in the management of a large project. Even though upper

management levels may not know or understand the trivial details, they can still monitor the general progress of a development team.

An itemized procedure is useful in teaching the process. It can provide a guideline and a structure that beginners can use to study a process such as the development of a computer system. The SDLC is often introduced in beginning level information systems textbooks to teach students what an information system is and how it is developed. This SDLC procedure can also be applied to the development of CAI with similar advantages.

As noted before, Table 2.3 lists the steps in the SDLC as defined by several writers. At first glance, the lists may seem to be substantially different. Looking for commonality and aligning similar steps horizontally on the table shows the similarity between lists. For example, each list begins with some sort of analysis or investigation that produces specific objectives and requirements of the system. Some lists include a separate preliminary investigation and a detailed analysis and some lists refer to the creation of objectives, identifying requirements and planning but they all are referring to basically the same processes.

The second major step relates to the process of designing the system. Some writers divide this into separate logical and physical designs. Other writers focus on the decisions that must be made during the design phase, which are evaluating and selecting alternatives.

Next, some sort of development and implementation is required. This involves purchasing any necessary hardware or software, creating any specialized software, testing the system, installing the system and commencing use of the system.

Finally, there is the need for evaluation of the system in terms of its objectives and continued maintenance to ensure it continues to meet those objectives. Figure 2.1 shows the common phases of all Systems Development Life Cycle models.

Table 2.3
Systems Development Life Cycle in Various Systems Textbooks

	Taylor Scientific Mgmt.	Cats-Baril Problm.-Solving	Burch SDLC	Capron Sys.Life Cycle	Cats-Baril & Thompson SDLC	Edwards Sys. Life Cycle	Gore & Stubbe SDLC	
1	Define the problem	Define objective Define requirements	Planning	Preliminary investigation				1
2	Identify alternative solutions	Generate alternatives	Analysis	Analysis	Analysis	Analysis	Study phase	2
3	Evaluate alternatives Select an alternative	Design system	General (conceptual) design Evaluation and selection Detailed (functional) design	Design	Design	Design	Design	3
4	Implement alternative	Implementation	Implementation	Development Implementation	Implementation	Development	Development	4
5		Evaluation	Maintenance		Maintenance and evaluation		Operation	5

Table 2.3 (continued)
Systems Development Life Cycle in Various Systems Textbooks

	Laudon & Laudon Sys. Life Cycle	Licker SDLC	Oz SDLC	Powers et al. SDLC	Shaw & Atkins System Project Activities	Shelly et al. SDLC	Whitten et al. SDLC	Wohl & Hunt SDLC	
1	Project definition	System study, problem def.	Planning	Investigation	Planning	Systems planning	Systems planning	Investigation	1
2	Systems study	Preliminary investigation	Analysis	Analysis and general design		Systems analysis	Systems analysis		2
3	Design	Logical design	Design			Systems design	Systems design	Design and development	3
		Physical design		Detailed design & implementation Installation					
4	Programming	Implementation	Implementation		Development	Systems implementation	Systems implementation		4
	Installation	Installation			Implementation			Implementation	
5	Post-implementation	Operation, maintenance,etc.	Support	System operation	Ongoing maintenance requirements	Systems operation and support	Systems support	Maintenance	5
				Review					

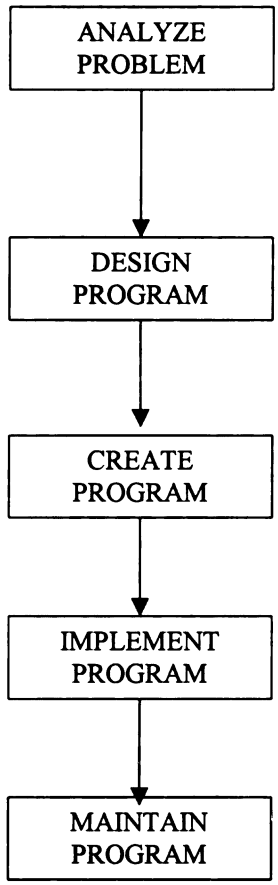

Figure 2.1. Common Phases of Systems Development Life Cycle Models

Diffusion of Innovation

In order to be implemented, any change must be diffused throughout the organization. This is true whether the change involves designing a new course, designing a component of a course, or introducing some technological innovation such as a computer application. Rogers (1995) has developed a six step model to illustrate the major phases of the "Innovation-Development Process." These phases are:

1. Needs/ Problems
2. Research (basic and applied)
3. Development
4. Commercialization
5. Diffusion and Adoption
6. Consequences (pp. 132-150)

While this model does not directly match the process models described earlier, there are many similarities. The process begins with the identification and analysis of a need or problem. Although Rogers describes research in the context of developing new innovative technology, that involves a type of analysis. Development relates directly to the earlier development phases, however much of the commercialization phase is new. Rogers specifically includes "production, manufacturing, packaging, marketing, and distribution of a product" in commercialization. The production part of the commercialization phase relates somewhat to activities included in the development phase of the earlier process models and distribution is typically considered a part of the implementation phase described earlier. Rogers' diffusion and adoption phase correlates to implementation. But the consequences phase requires closer examination. Rogers defines this as "the changes that occur to an individual or to a social system as a result of the adoption or rejection of an innovation" (p. 150). It is in his further elaboration that you can see this relates to the evaluation phase for he says, "here the original problem/need that set off the innovation decision process is either solved or not."

To Rogers, innovators are individuals within a system who are the first to adopt innovations. While he does not specifically exclude the possibility that an insider could have created the innovation, he perceives the innovator's role as "launching the new idea in the system by importing the innovation from outside of the system's boundaries" (p. 264). This clearly indicates that Rogers does not expect the insider to be the developer of

the innovation and that he perceives the innovation as being created by an outside developer.

Similarly, Rogers introduces the role of a change agent whose role is to "facilitate the flow of innovations from a change agency to an audience of clients" (p. 336). Throughout his discussion, the change agent is also presented as someone outside of the "client system" who must communicate and persuade the clients. The change agent is definitely an outsider.

The size of an innovation, component versus entire system, is also not a factor that Rogers is interested in. He does identify five attributes of an innovation that affect diffusion, but these are relative advantage, compatibility, complexity, trialability and observability. Even complexity refers not to anything related to size but the perception that an innovation is "relatively difficult to understand and use" (p. 242).

It should also be noted that there has been some interest in the introduction of change into educational institutions in particular. Havelock (1973 and 1995) lists six stages of planned change in education (Table 2.4). Generally, stages two to six of the procedure for introducing change parallel the steps in the Systems Development Life Cycle. Interestingly Havelock does not list development as a separate stage. In stage four he does refer to research and development (R & D) but says, "the *less* adaptation you have to do, the better off you are. This is why it is so important to be a good utilizer, taking maximum advantage of the R & D efforts of others" (p. 109).

Table 2.4
Comparison of Change Agent and SDLC

How the Change Agent Works (Havelock)	Systems Development Life Cycle (SDLC)
(1) Building a relationship (between change agent & client)	
(2) Diagnosis: From pains to problems to objectives	Analysis
(3) Acquiring relevant resources	Design and Development
(4) Choosing the solution	
(5) Gaining acceptance	Implementation
(6) Stabilizing the innovation and generating self-renewal	Maintenance

In stage five, gaining acceptance, Havelock (1973) presents a list of six phases or stub-steps in the adoption process:

a. Awareness
b. Interest
c. Evaluation
d. Trial
e. Adoption
f. Integration (pp. 113 - 114)

The change agent must help the client pass through these 6 phases in order for the change to become self-sustaining and a part of routine, true integration. This list helps describe the final steps before reaching stage 6: Stabilizing the Innovation and Generating Self-Renewal.

Generally Havelock does tend to separate participants into change agents and clients. While this seems to imply the change agent is an outsider to the client organization, he does address the possibility that the change agent could be an insider. In fact, he has an entire section in which he compares the advantages and disadvantages of

an inside change agent versus and outside change agent. I have summarized and compared these in Table 2.5.

Table 2.5
Comparison of Inside and Outside Change Agents

INSIDER ADVANTAGES	OUTSIDER DISADVANTAGES
Knows system	The outsider may lack the knowledge of the insider
Speaks the language	
Understands the norms	
Identifies with the system's needs and aspirations	May not "care enough"
Is a familiar figure	Is a stranger
INSIDER DISADVANTAGES	**OUTSIDER ADVANTAGES**
May lack perspective	Is in a position to have perspective
May not have special knowledge or skill	Is in a position to bring in something genuinely new
May have to live down his past failures	Starts fresh
May not have an adequate power base	Is independent of the power structure
May not have the independence of movement	
The inside change agent usually faces the difficult task of redefining his on-going relationships with the other members of the system	

Relative to the insider disadvantage that "may not have the independence of movement," Havelock adds, "the obligations of membership may severely limit the time and energy that he can invest in his new role" (p. 51) as change agent. This was a key difficulty in this case study. In comparing the insider and the outsider, Havelock suggests that one alternative might be "a 'change agent team' in which both insiders and outsiders work together" (p. 53). This way, the team members can "try to maximize the strengths of both positions in the service of innovation" (p. 53).

Havelock's major focus is on the change process. Although he does examine some features of a potential solution, such as costs, benefits and workability, he does not

address the severity, completeness or size of a change. Therefore, he makes no comments on differences between creating or changing an entire course versus creating or changing one element of a course.

Altamirano (1998) used Rogers' and Havelock's viewpoints to examine the role of a teacher in the change agent and client relationship. She studied an individual teacher who incorporated a computer program into her teaching. Altamirano defined the teacher as a client who found and used technology. Altamirano also defined the teacher as a change agent who "introduced the technological innovation to her students and helped them use it as a means to learn about the course's subject matter" (p. 11). As in this study, Altamiarno found that prior knowledge and experience were extremely important. She stated that both diffusion processes, where the teacher was client and where the teacher was a change agent, "were influenced highly by what the teacher brought with her to technology use" (p. 211).

Strudler and Gall (1988) examined computer coordinators as inside change agents. They found that "while effective coordinators must be knowledgeable about instructional computing, it is just as critical that they possess strong interpersonal and organizational skills" (p. 7). Although inside computer coordinators have knowledge of the organization's system, language and norms, they are still a stranger or outsider to a teacher. "Teachers appear especially resistant to 'computer experts' who 'know, but can't teach'" (p. 7).

Summary

The roles of instructional design, technology, and diffusion of innovation in education are becoming increasingly important. As technology grows, the number of options, features and possible uses in education increase. Unfortunately, the technology also tends to become more complex, encouraging a trend towards specialization and the need for development teams and the need for more "systematic, systemic planning. Most educators have little background and fewer resources with which to conduct that kind of planning" (Lawler, et al., 1998, p. 29). Even task forces of insiders are sometimes insufficient. "Task forces hold considerable local knowledge but, because innovations are often unfamiliar, local people may not have enough technical knowledge about the contemplated change. Conversely, consultants can provide needed technological expertise but typically lack awareness of the organization's unique needs" (Wilson, et al., 1999, p. 25).

More and more theorists and researchers are examining the importance of involving the teacher in the design and development of instruction, educational uses of technology especially computer systems, and diffusion of change in the classroom. Wilson (1999) states "end users need to think like designers as well as consumers" (p. 16). In fact, teachers as end users "always function as designers as they appropriate and use learning resources, even if they are left entirely out of the formal design of those learning resources" (p. 15) As others have, Moallem and Earle (1998) conclude that "the image of teachers as designers of their own instruction needs to be emphasized in the instructional technology field" (p. 17). This study builds on that background and examines how the principles of instructional design, education and technology, and

diffusion of innovation were involved in the design and development of a computer instructional aid by an instructor as one new component that could be used in a multi-section course that that instructor teaches.

CHAPTER THREE

METHOD OF THE RESEARCH

Introduction

This study is to examines the development and diffusion of a computer instructional aid. To achieve this purpose I, as an insider, developed a computer instructional aid for a multi-section course, disseminated it and assessed its usefulness by teachers and students. Therefore, I needed to examine four major stages: (1) the preliminary development of the computer instructional aid when I conceived and tested the feasibility of creating the aid; (2) the actual development of the aid; (3) the dissemination, implementation and testing of the aid in the fall of 1998; and (4) the period and events that followed that implementation. But first I will describe the environment in which all of this took place.

Context of the Development

Baker College

This research was conducted at Baker College, a private not for profit post-secondary institution. The Baker College schedule is based on a ten-week quarter. It grants Associate and Bachelor degrees in the areas of Business Administration, Computer Information Systems, Office Administration/Secretarial, Health and Human Services, and Engineering and Technical. Baker College also offers a Masters degree in Business Administration. Baker College offers courses at eight campuses and more than six additional locations throughout the state of Michigan (see Figure 3.1). The stated mission

of Baker College is "to provide quality higher education which enables graduates to be successful throughout challenging and rewarding careers."

Figure 3.1. Baker College Campuses

The Baker College System is one the largest private post-secondary educational institutions in Michigan. Although different programs may be offered at different campuses, every attempt is made to standardize the courses within each program and the content of each course throughout the system. This minimizes student difficulties in progressing from one course to another in a sequence, even if different instructors teach those courses. This standardization also minimizes student problems in transferring between campuses in the Baker College System. This also means that essentially all

students on all campuses take the same beginning general- education courses with the same content. As a result, instructors can more easily share instructional techniques and aids, such as a computer instructional aid.

Baker College was a good place for this study because there were so many sections of one specific course where this computer instructional aid could be used. With the potential of being used by so many students, the amount of time and effort to develop the instructional aid could more easily be justified. In addition, the common final exam clearly defined and limited the content that needed to be covered by the aid.

Course

The course involved in this study was Information Systems Theory (INF111). This course is taught in multiple sections on all campuses (see Table 3.1). The Baker College schedule is based on a ten-week quarter. The course is 20 hours long (two credit hours) and meets for four hours each week for half of the quarter, five weeks. This is considered a computer literacy course and is classified by the college as a general education course. It consists primarily of terminology and definitions that must be learned by the student in order to understand what is taught in other computer courses. The course content is determined by a list of course outcomes (Appendix D) which is approved by a system committee that includes department deans and instructors from all campuses. Each of the five major course outcomes is stated in general terms but is intended to cover all major related topics. For example, the third course outcome is

> Terminology and principles related to data and data storage
> in computer information systems including:
> a. Data to information processing cycle.
> b. Data hierarchy and organization.

c. Batch and interactive processing.
d. Coding and number systems (ASCII, binary, etc.)
e. Measurement of storage space (bit, byte, etc.)
f. Features of data storage including volatile and nonvolatile, memory addresses, and RAM vs. ROM.

In the fall of 1994, the previous two four-credit introductory computer courses that most students were required to take were divided into four two-credit courses. The theory, terminology and concepts were placed into the INF111 course. The other three classes were hands-on classes teaching word processing, electronic spreadsheets, and file management. By fall of 1995 a waiver test was developed for those students who had prior computer experience. If a student scored 80% or better on the waiver test, he or she did not need to take INF111. Eventually this waiver test was also adopted as a common final for all sections of this course taught in the Baker College System.

The waiver test currently consists of 81 multiple choice questions. For example, question number 53 is:

Approximately one million bytes is referred to as a _____.
A. Decabyte
B. Kilobyte
C. Gigabyte
D. Megabyte

A committee regularly evaluates the exam to be sure it covers current important computer terms and concepts. The last major revision was in the summer of 1998. At that time questions that were considered confusing or out-of-date were dropped and some questions that covered new important topics were added. For example, a question that was dropped because it was out-of-date since all monitors are now SVGA and the other types are no longer available was

The resolution of most monitors sold for personal computer systems is _____.

A. CGA (Color Graphics Adapter)
B. EGA (Enhanced Graphics Adapter)
C. VGA (Video Graphics Array)
D. SVGA (super VGA).

A question that was added was

The speed at which the entire screen is redrawn is called the _____ rate.
A. Resolution
B. Refresh
C. Pixel
D. Diffusion

Faculty

Because this course is considered a general education course, the state requires that it must be taught by an instructor with a masters degree. Most of the instructors teach at least one other "hands-on" beginning computer course. Typically, most of the instructors who teach this course are part-time. Of the 32 instructors teaching this course in the fall of 1998 (see Table 3.1), two were full-time instructors and one was a full-time administrator/instructor. All of the other instructors were part-time.

Table 3.1
Campuses, Students, Sections and Instructors in Fall 1998

Fall, 1998		INF111- Info. Systems Theory		
		as of 9/16/1998		
Campus	Total Students	Students	Sections	Instructors
Auburn Hills	1319	140	9	4
Cadillac	801	88	4	4
Flint	4112	356	11	5
Jackson	1010	127	8	6
Mount Clemens	1630	192	7	4
Muskegon	2370	207	10	5

Owosso	1847	148	5	2
Port Huron	1036	99	4	2
Totals	14125	1357	58	32

Two instructors were involved in the early stages of this study, (1) the conception and the feasibility stage; and (2) the actual development of the program. These instructors were Lori, a full-time instructor on the Owosso campus; and Joyce, a full-time instructor on the Port Huron campus.

Four instructors and six sections were involved in this study during the implementation and testing of the aid, stage 3: the five-week period from September 21, 1998 to October 23, 1998. The instructors were Joyce, the full-time instructor on the Port Huron campus mentioned above; Judy, a part-time instructor on the Owosso campus; Tina, a part-time instructor on the Muskegon campus; and Cynthia, a part-time instructor on the Flint campus. (See Table 3.2)

I used three major criteria for choosing these instructors. First, I wanted as many campuses represented as possible. I wanted to minimize any possible student differences between campuses and I hoped that it would be easier to disseminate the aid to other instructors if they could talk to an instructor on their own campus that had used the aid. Second, I wanted instructors with prior experience with this course so they could compare the sections that used the computer instructional aid with prior quarters that did not have the aid. I also hoped that since these instructors would be familiar with the problems with this course they could give me valuable insight as to whether or not the aid helped. Third, I needed instructors that would be open minded enough to give me precious class time to distribute, explain and demonstrate the computer instructional aid.

All of these instructors had taught this course at least once before. Joyce and Tina had taught it many times. All of them were also well-known to the other INF111 instructors on their campuses and therefore could be influential in encouraging other instructors to use the aid.

Students

Students taking Information Systems Theory (INF111) have a high school diploma or a G.E.D. degree so they are usually 18 years of age or older. Almost all students at Baker College must take this course unless they have transfer credit for a similar course from a post-secondary institution or passed the waiver test for this course.

During the fall quarter of 1998, 1357 students took this course in 58 sections of this course on 8 campuses. Data was collected from 6 sections of this course and a total of about 136 students. (See Tables 3.1 and 3.2) Drops, absences and schedule changes caused the difference in number of students registered as of September 16, 1998 and the number of computer instructional aids eventually distributed. Later, I will talk about the reason for the differences in students registered and aids distributed.

Table 3.2
Sections & Instructors in Study as of 9/16/1998

Campus	Instructor	Section	Time	Students Registered	Aids Distributed
Flint	Cynthia T.	930	6-10 M	40	24
Muskegon	Tina N.	370	10-12 TTh	26	18
Owosso	Judy R.	2495	10-12 MW	33	29
		2510	4-6 MW	28	25
Port Huron	Joyce N.	2310	8-10 MW	24	21
		2315	10-12 MW	24	19
Totals				175	136

Faculty Developer, "The Insider"

Because I am both author and the major participant in this study, it is important for me to describe the background that qualified me as the "inside developer." I originally started as an accounting instructor at the Flint Campus. In 1980, on year after becoming full-time, I agreed to take additional training so I could become a computer instructor. In 1983, after taking one college course and several training courses, I started working part-time on a masters program at Eastern Michigan University, which I completed in 1987. This program was in the Educational Psychology department of the College of Education and focused on educational technology. One of my major purposes was to learn more about computers and technology. But I was also interested in learning more about learning and teaching so I could become a better teacher. During this period I also became interested in instructional design and started volunteering to design new courses and major changes to existing courses when the opportunities arose.

From 1980 to 1985, when I moved to the Owosso campus, I taught at least one introductory computer course each quarter. There were a few quarters after that when I did not teach an introductory course. Prior to fall of 1994, I taught over 40 sections of the four-credit introduction to computers course.

In 1988 I started a Ph.D. program at Michigan State University in Educational Systems Development in the Counseling, Educational Psychology, and Special Education department. I mainly focused on educational technology, instructional design and adult learning.

In fall of 1994 the current two-credit introductory computer course was introduced. That quarter I volunteered to "pilot" teach several sections of the course so that I could develop materials and tests that I could share with other instructors on our campus. Prior to the fall of 1998, I taught about 10 sections of the two-credit introduction to computers course. In recent years, I usually only teach this course about once during the regular academic year and once during the summer but I have been involved with all of the committees that made decisions regarding content, textbook, and the waiver test. I have also taught many computer programming courses and courses concerned with the development of computer systems.

Overview of the Development Process

I am presenting this overview of the process to explain the development of the computer instructional aid and to clarify the methods used to collect information and to document this process. I have divided this overview into four major time frames: (1) preliminary development of the computer instructional aid; (2) actual development of the computer instructional aid; (3) dissemination, implementation and testing of the aid; and (4) period and events that followed the implementation. (See Figure 3.2) Following the overview I will elaborate on the methods used to document activities in each time frame.

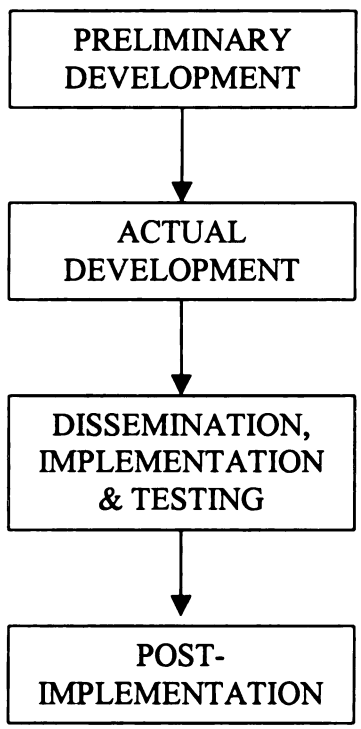

Figure 3.2. Overview of Actual Development Process

Preliminary Development of the Computer Instructional Aid

While working on my masters degree at Eastern Michigan University, I created a simple question and answer program that would (a) present the student with a question; (b) accept the student's answer; (c) compare the student's answer with the correct answer; and (d) provide feedback to the student as to whether the answer was correct or incorrect. While teaching INF111, the multi-section course used in this study, during the academic years 1994-95 and 1995-96, I was looking for better ways to help my students learn the material they needed to know and remembered the earlier computer program I had written. This inspired me to consider creating a similar program that would assist INF111 students in reviewing for their final exam and I began this project. During this period, I informally discussed the possibilities of this project with two experienced INF11 instructors, Lori and Joyce.

Actual Development Of The Computer Instructional Aid

Since the computer instructional aid was a computer program, I followed the typical computer system development process in creating the computer instructional aid to help students review for their INF111 final exam. This involved (a) analyzing the problem, (b) designing the program, (c) creating the program, (d) implementing the program, and finally (e) maintaining the program. (See Figure 3.3) These five steps were obtained by examining many textbooks dealing with computer systems analysis, design, methods and development. In most textbooks this computer system development process is referred to as the System Development Life Cycle (SDLC). Table 2.3 shows the chart

that was used to compare the steps listed in 15 different references to derive the above five steps.

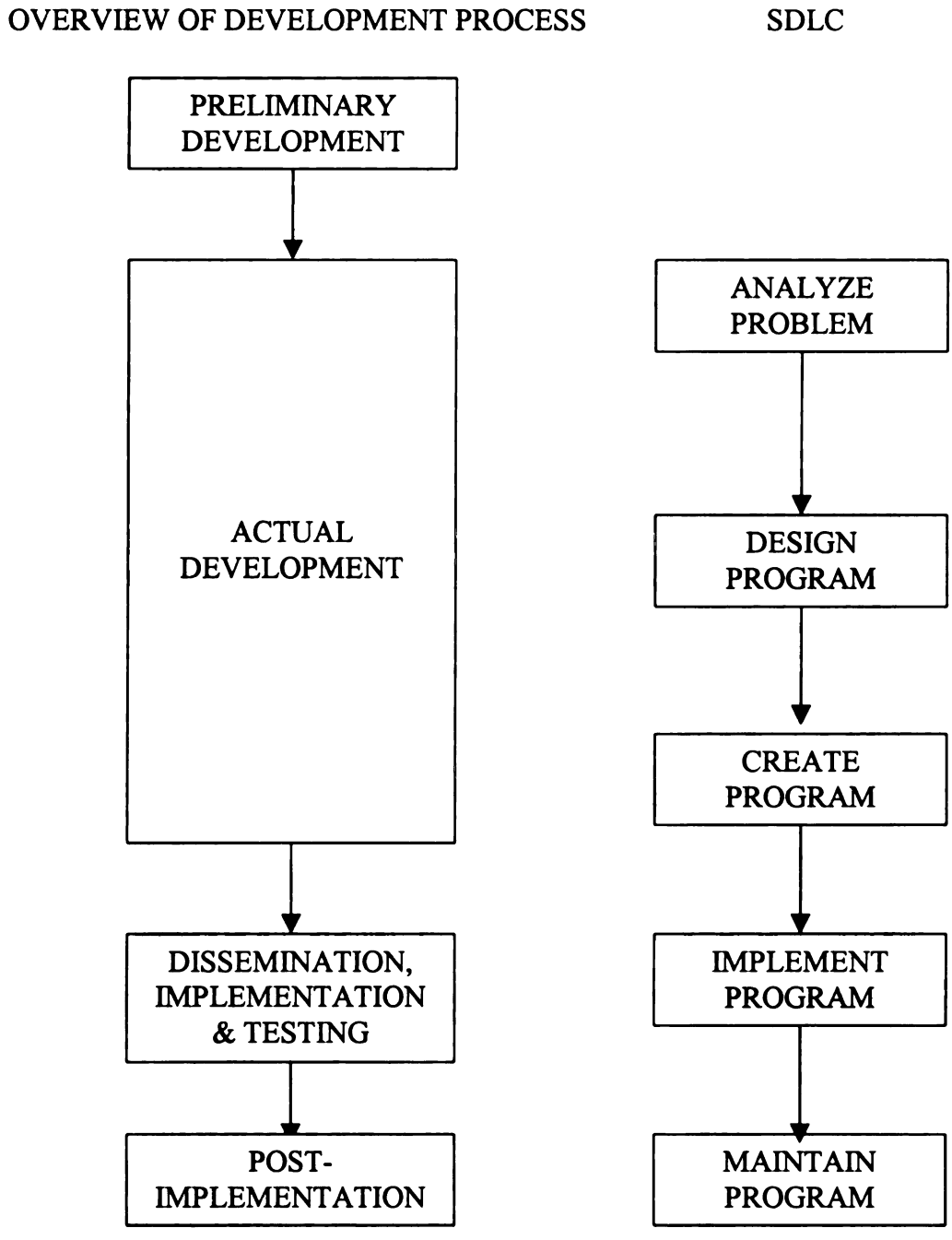

Figure 3.3. Comparison of Development Process and SDLC

My overview of the development process combines the first three steps of the SDLC (Figure 3.3) because I used a variation of the SDLC called prototyping. This approach involves (a) designing and creating a portion of the computer program, (b) comparing it with the requirements, (c) identifying one or more necessary modifications, (d) designing and adding these modifications to the program, (e) and repeating steps (b), (c) and (d) until no additional modifications are identified when the program is compared to the original requirements (Figure 3.4).

During the actual development process, I also combined the SDLC (Systems Development Life Cycle) model with Dick and Carey's instructional design model for developing instructional materials (1996). Figure 3.5 matches their eleven steps with the five SDLC steps I derived.

In identifying the requirements for the drill-and-practice program, which was the computer instructional aid, I relied heavily on the common final exam, the course outcomes, my personal experiences, and the experiences of two other experienced instructors. The next two phases, design and creation, were actually done together since I used a prototyping approach. I used portions of the earlier program as a guide. I saved and printed out each programming "try" so that I could later review my problems, progress and decisions in this process. The creation process involved coding, entering, running and testing each version or "try." During this stage, the two experienced instructors tested one of the final versions for me. As part of a "pilot" study, these instructors also gave it to some of their students and let them test it.

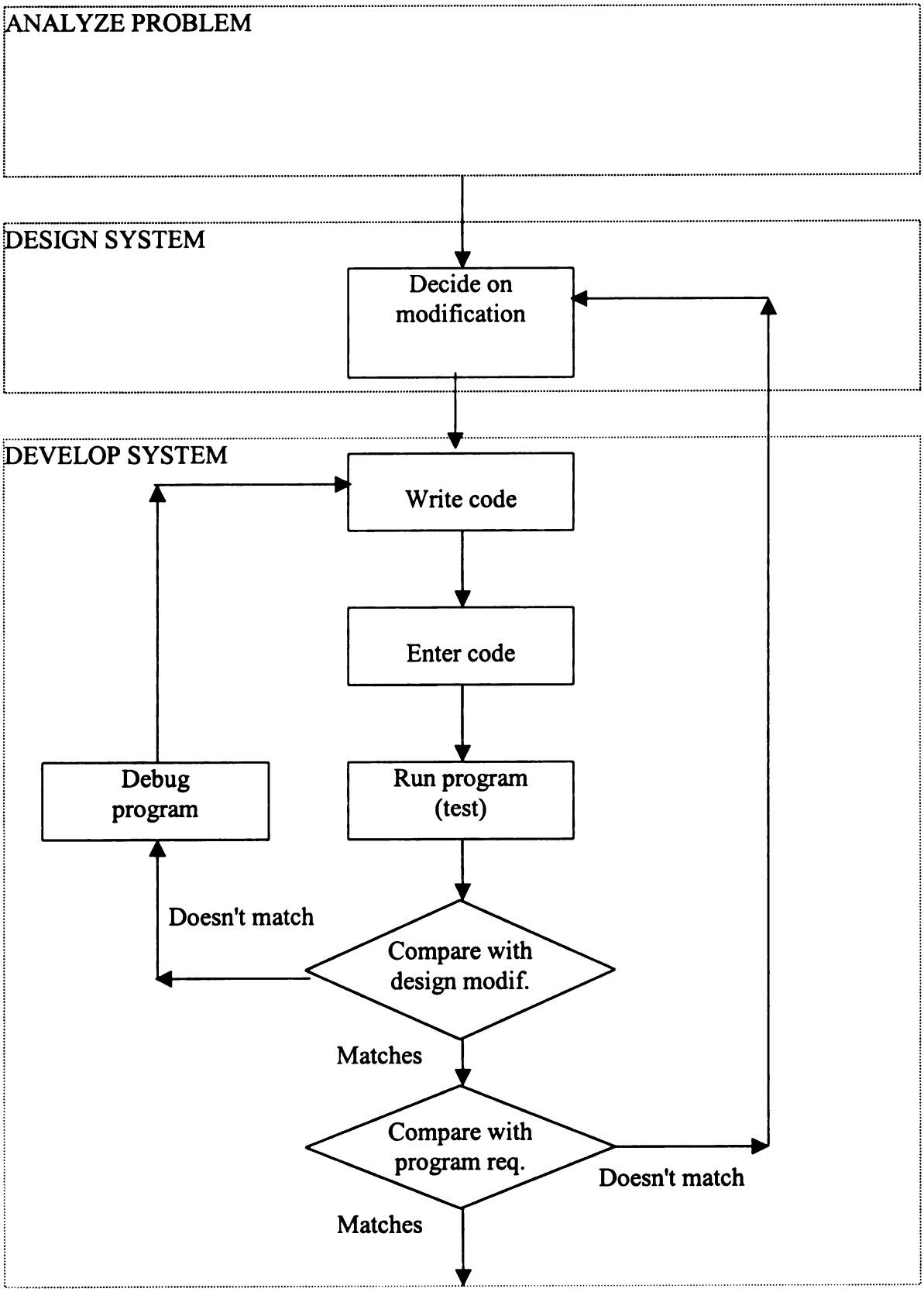

Figure 3.4. Prototyping Model

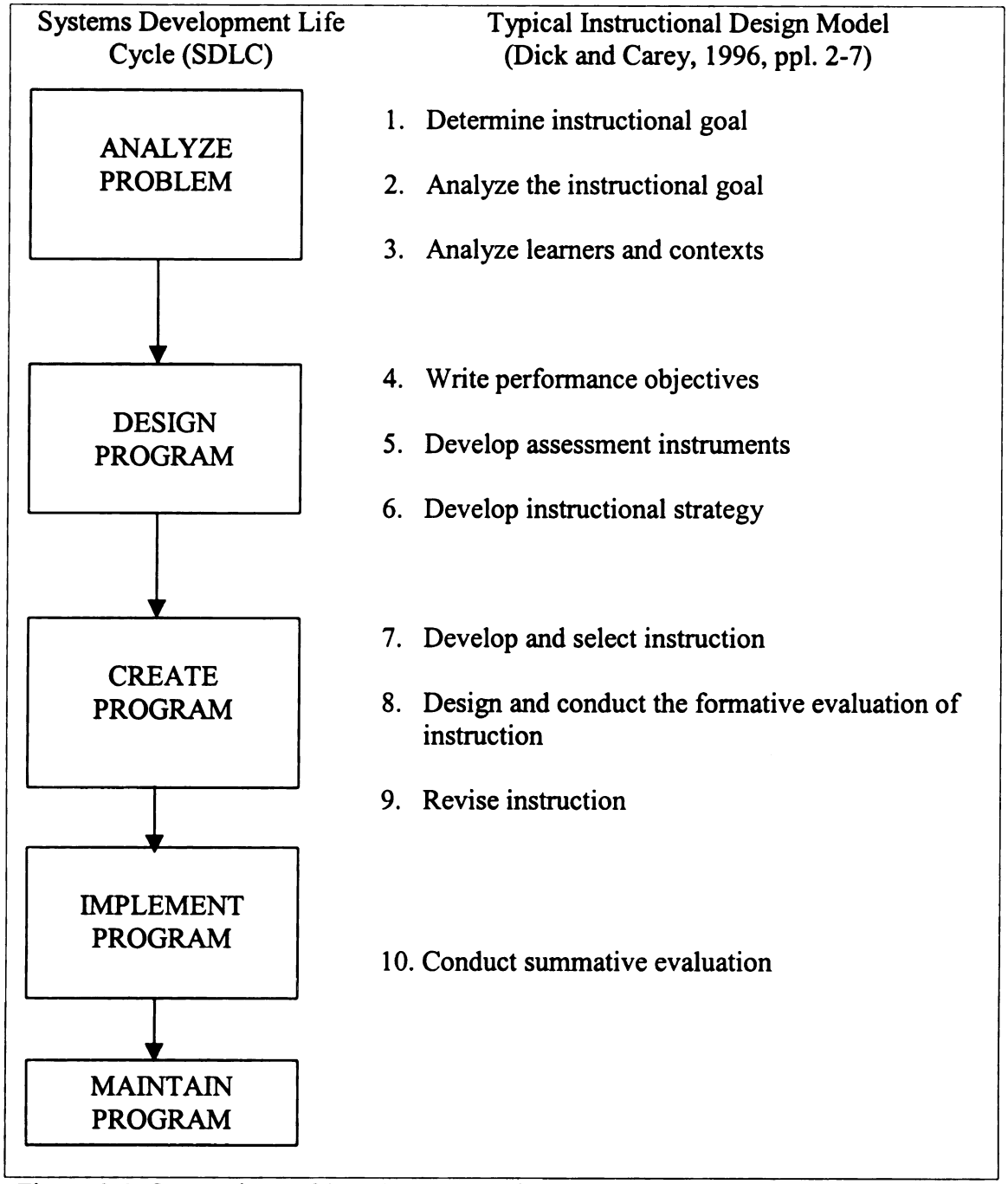

Figure 3.5. Comparison of SDLC and Steps in Developing Instructional Materials

Dissemination, Implementation and Testing of the Aid

This third stage coincided with step (d) of the SDLC process described above: implementing the program. Prior to the start of the fall 1998 quarter, I contacted four instructors, described above, and made arrangements to distribute the computer

instructional aid on floppy disks to their students. At the end of the course sections that received the program, I attended the final exam, collected the floppy disks, distributed a short questionnaire before the exam, and collected it after the exam. I then interviewed the instructors after the course. I planned on using interviews with the instructors and the results of the student questionnaires to assess the usefulness of the tool. (Figure 3.6)

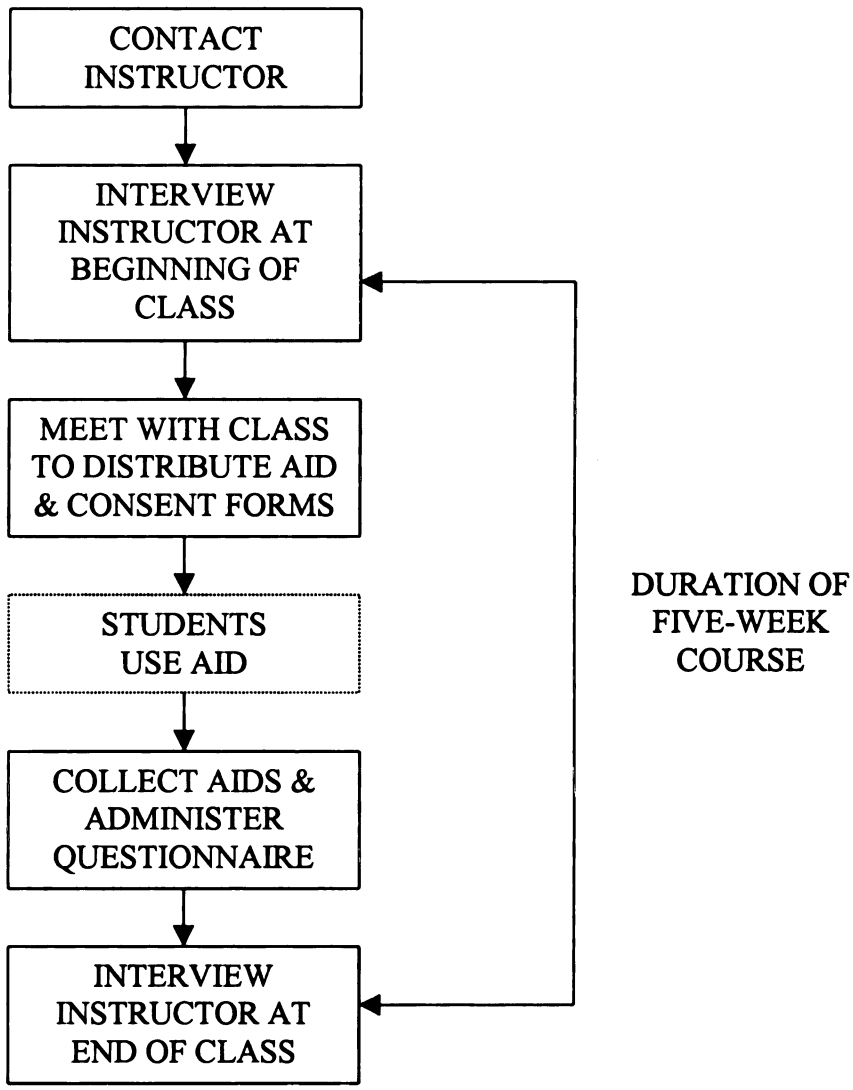

Figure 3.6. Steps in Dissemination, Implementation and Testing of the Aid

Period and Events that Followed the Implementation

After completing the above stages, I encouraged the instructors to continue using the computer instructional aid in their courses. During this final stage it was necessary to make some minor changes in order to adjust the program to coincide with a new version of the textbook. Such changes are referred describe as maintenance, step (e), in the SDLC. I also attempted to distribute it more widely to other instructors and campuses.

Method of Collecting Information

I divided the development process into four major time frames: (1) preliminary development of the computer instructional aid; (2) actual development of the computer instructional aid; (3) dissemination, implementation and testing of the aid; and (4) period and events that followed the implementation (Figure 3.2). I used these four time frames to explain the development of the computer instructional aid and I will now use them again to describe the methods of collecting data during each of these time frames.

Collecting Information During Preliminary Development

To document the earliest phases of development I have the printout of the earlier computer program, some informal notes, my recall of the situation and events, and the recall of Lori and Joyce, the Baker College System CIS (Computer Information Systems) committee meeting minutes, copies of the different versions of the waiver test and notes from subcommittee meetings to revise the test.

Collecting Information During Development of the Computer Instructional Aid

While I was developing the computer instructional aid, I had several additional conversations with the two experienced instructors I consulted, Lori and Joyce. I have notes from two fairly formal conversations and an audio tape of one of these conversations. During the academic year during which I developed this aid, Lori and I shared an office. About once every two to four weeks we casually talked about this course, its content, student difficulties, and various instructional aids that were being used. Joyce is also a good friend of mine and on a couple of occasions during the preliminary development we informally discussed my ideas. I have no notes on these informal conversations so I rely on my recall of them. However, I checked the gist of these conversations with Lori and Joyce and recorded any differences in remembrances.

I have a copy of each major version of the computer program saved on a floppy disk and printed out. I also have notes on those printouts describing the problems and showing the changes I was planning on making. At the end of the pilot study I informally interviewed Lori and Joyce. I have notes from the interview with Lori and an audio tape of the interview with Joyce.

Collecting Information During Dissemination, Implementation and Testing

Prior to the fall quarter of 1998, Joyce agreed to give the waiver test the first day of class in two sections. She and I have done this before to determine if there are any students who can waive the class. Since the common final is the same as the waiver test, this also provides students with a general idea of the material they need to learn in this class. Some students, with some prior computer experience, are not convinced they need

the class until after they taken and failed to pass the waiver test. For purposes of this study, this also provided a pretest to determine prior knowledge at least in these two sections. This seemed a reasonable check of what students would know. There is no reason to think students in other sections could be any different for pretest.

I interviewed all four instructors before the course began (Table 3.3). One purpose of this interview was to identify instructor opinions and attitudes about the course. For example, the first question was "Do you believe this is an important class? Why or why not?" Another purpose of the interview was to determine the instructor's attitude about the common final exam and helping students succeed on the final exam. Question number four was "What aids and tools do you use to help students pass the final exam?" In the interview I also identified any changes or modifications that the instructor might like made to the computer program. For example, question number eleven was "What results, if any, would you like the program to print out when the student ends the program?" A complete list of interview questions is attached in the Appendix L and Table 3.3. I took notes and have audio tape on three of the four interviews. In lieu of an interview with Cynthia she sent me written answers to my interview questions via email. I can document interview dates from my calendar and dates on consent forms.

Table 3.3
Instructor Interview Questions at Beginning of Course

INSTRUCTOR INTERVIEW AT BEGINNING OF COURSE
1. Do you believe this is an important class? Why or why not?
2. What do you feel should be the most important part of this class?
3. Are you satisfied that the common final appropriately tests what is important in this class?
4. What aids and tools did you use to help students pass the final exam?
5. Do you think a computer program that presents questions from a test bank might help your students review and prepare for the final?
6. How difficult is it to use a computer room during class time?
7. May I use some class time to distribute this program and demonstrate it to your

students?
8. Would it be possible to require your students to turn in the printout indicating they used the computer program?
9. What questions would you like in the test bank used by the computer program?
10. Do you think the student should be able to print out the questions?
11. What results, if any, would you like the program to print out when the student ends the program?
12. Is there anything else you would like the program to do?
13. What problems and concerns do you have regarding the use of this computer program in your class?

I was able to attend a class session of each section sometime during the first half of each five-week period. Again, I documented dates from my calendar and dates on consent forms. At that time I handed out floppy disks containing the computer instructional aid and an instruction sheet telling students how to run the program. In Judy's sections and Joyce's sections, I was able to work with the students in a computer room. In Tina's section and Cynthia's section I was able to demonstrate the aid with a computer and LCD monitor projector.

The floppy disks that I distributed contained the computer program that was the computer instructional aid. In addition to serving as an instructional aid, the program also recorded some data regarding the student use of the program. The computer program made an entry in a separate data file each time the student ran the program, selected a topic and textbook chapter from the main menu, looked at one or more questions for that chapter, and then exited that group of questions. This entry contained the student name, the chapter number, the number of questions attempted, the number of questions answered correctly, and whether or not the student examined all questions for that chapter. Tina's section and Cynthia's section met the last five weeks of the ten-week quarter. All of the other sections met the first five weeks. Those two sections received a

slightly modified version of the computer instructional aid that also recorded the date and approximate time that the questions from each chapter were attempted.

I attended the final exam of all of the sections involved in the study except for section 2495 (Owosso, Judy, 10-12 MW). I distributed a questionnaire to all students when the final exam was handed out (Table 3.4). I announced that I would appreciate it if they would complete the questionnaire after they had finished the exam but made it clear that this was completely voluntary. I also asked them to return the floppy disks, containing the program, to me. Because of scheduling and traffic problems I was late in arriving at one of Judy's sections. As a result, I received five surveys from that section. However, I did get almost all of the disks returned. After the instructors graded the final exams, they sent me the final exam grades for all students who had signed the consent forms.

Table 3.4
Student Survey

INF111 REVIEW PROGRAM
STUDENT SURVEY

STUDENT NAME:_____
CAMPUS : _____ DAY(S): _____ HOUR: _____

1. How useful was the software?

Very Useful	Somewhat Useful	A Little Bit	Not At All	Not Applicable
(1)	(2)	(3)	(4)	(NA)

SUGGESTIONS FOR MAKING THE PROGRAM MORE HELPFUL:

2. How easy was the software to use?

Very Easy	Easy	Somewhat Difficult	Very Difficult	Not Applicable
(1)	(2)	(3)	(4)	(NA)

PLEASE DESCRIBE ANY PROBLEMS YOU HAD:

3. Why did you use the program? What influenced you?

4. Did you receive adequate instruction in how to use the program?

Very Good	Good	Adequate	Inadequate	Not Applicable
(1)	(2)	(3)	(4)	(NA)

COMMENTS:

5. Where did you use the program? Check all that apply.

___ home
___ at a friend or relative
___ work
___ open computer lab at Baker College
___ LSS
___ other Please describe:

6. The software covered 7 chapters. Please check all chapters you used:

___ 1. An Overview of Computer Concepts
___ 2. Computer Software Applic.: User Tools
___ 3. Input to the Computer
___ 4. The System Unit
___ 5. Output from the Computer
___ 6. Secondary Storage
___ 7. Communications and Networks

7. Think back over all of your use of the software. How many hours total did you use the software?

COMMENTS:

After the course ended, I asked instructors some additional questions (Table 3.5). These questions were used to determine if instructors believed the computer program was a beneficial component of the course. The questions also helped find factors that influenced the usefulness of the computer program, as perceived by the instructor. For example, the first question directly asked "Did you like using the computerized questions and answers as a student study aid?" A complete list of questions is attached in the appendices. I interviewed Judy and Joyce and have audio tapes of those interviews. Because of scheduling problems and conflicts, Cynthia hand-wrote answers to the questions and Tina emailed me her answers.

Table 3.5
Instructor Interview Questions at End of Course

INSTRUCTOR INTERVIEW AT END OF COURSE
1. Did you like using the computerized questions and answers as a student study aid?
2. Why do you feel this was or was not a helpful tool for this course?
3. What problems did you encounter with the program?
4. What features of the program did you like?
5. What features of the program did you NOT like?
6. Any suggestions to make the program better?
7. Did you give students credit for using this program?
8. Would you like to use the program the next time you teach this course?
9. Do you think a similar tool would be helpful for other courses?

Collecting Information During the Period that Followed the Implementation

In spring of 1999, I contacted each of the teachers and asked them if they had used the computer instructional aid in any later classes. I contacted all of them by phone or email and received responses from all of them. I took notes of these conversations.

I also submitted the computer instructional aid to the Baker College System curriculum

office. I sent a copy of the review program, a copy of the GW-BASIC program which is required to run the review program, a set of instructions for the student, and a set of instructions for the instructor.

In fall of 1999, I sent an email message to all of the CIS deans informing them that I had this computer instructional aid. I asked them to inform their INF111 teachers and if any were interested in using it, to contact me.

Summary

During 1998, I designed and developed a computer instructional aid to help INF111 students at Baker College practice for a required common final exam. This is a

multi-section course taught on all campuses by many instructors, both part-time and full-time that focuses on introductory level computer terminology and definitions. I carefully documented the development process. I tested the aid by doing a pilot study during the spring of 1998 with three instructors, including myself. I then formally implemented the aid in the fall of 1998 with six sections of the class taught by four instructors on four different campuses. I interviewed the instructors before the implementation and after the implementation. I distributed to and collected survey forms from students in those sections and also obtained their scores on the common final exam. For two sections I obtained data the computer instructional aid saved on the floppy disks the students used.

CHAPTER FOUR

RESULTS

The results of this study are the documentation of the detailed process of an inside developer of a computer instructional aid. These results will be compared with other process models in chapter five. I will present the process results in a chronological manner, starting with the preliminary development of the computer instructional aid. I will divide the process results related to this preliminary development into three cases to present the data obtained from the two instructors that worked with me during this period and myself.

Then I will present the process results related to the actual development of the computer instructional aid, dividing that section into the five phases of the Systems Development Life Cycle (SDLC) described in chapter three: analysis, design, creating the program, implementing the program, and maintenance.

Next I will divide the section on implementing the program into four cases dealing with the data obtained from each of the instructors and their classes during this phase. For each case I will include data from the instructor interview at the beginning of the course, data from the student survey at the end of the course, and data from the instructor interview at the end of the course. I will then describe the data obtained during the evaluation of the implementation phase from all four cases.

To conclude, I will present data obtained after the development of the computer instructional aid including data related to the final phase of the SDLC, maintenance.

Preliminary Development of the Computer Instructional Aid

As explained in chapter 3, in January, 1988 while working on my masters degree at Eastern Michigan University, I created a program in GW-BASIC (Appendix A) that imitated the Pilot programming language by reading lines from an ASCII text file. That program then performed different actions based on special character combinations at the beginning of some lines. The text file, as in a Pilot program, could contain messages that would be presented to the user; allow the user to enter responses, such as to a multiple choice question; compare the answer entered by the user to a stored "correct" answer; provide alternative feedback messages to the user based on whether or not the user's answer matched the "correct" answer; and then present the next messages in the text file.

During the summer and fall of 1989 I created text files, to be used with the above program, containing practice multiple choice questions for my introduction to computers classes (Appendix B). After the student entered a chapter number, the program would load the appropriate text file and then present one multiple choice question after another to the student. After each question the student would enter his or her answer, the program would display a "CORRECT" or "INCORRECT" message, and then the next question would be presented to the student. After the program had presented all questions for that chapter to the student, the program would end. There was no randomization, students could not exit until all questions for that chapter had been presented, questions answered incorrectly were not represented to the student and the student had to restart the program in order to redo that chapter's questions or look at questions from another chapter.

In July of 1990, I used a variation of this program as part of a homework assignment in a graduate class on adult learning at Michigan State University in the summer quarter of 1990 (EAD 861). We were given some material to present to the class and encouraged to use creative methods for presenting that material. I used the program described above to present a portion of the material; ask a multiple choice question; repeat the material if the user's answer was incorrect; or present the next portion of material if the user's answer was correct (Appendix C). I also added a feature that let the student restart from the beginning after all text and questions had been examined without restarting the entire computer program. I also experimented with more descriptive feedback than just "CORRECT" or "INCORRECT." I tried to make the responses more conversational, such as "Yes, keeping accurate records is most important" and "No, while that is important, keeping accurate records is most important."

While teaching INF111, Information Systems Theory, in 1994-96, I started considering the possibility that a similar program might be useful to my students. The major influence was the waiver test that I helped develop in 1995. From the beginning, the Baker College System CIS (Computer Information Systems) committee considered the possibility of using the waiver test as a common final exam. The CIS committee decided it would be best if the exam contained between 80 to 100 multiple choice questions, since administrative requirements limited the waiver exam to less than one hour and needed it to be easy to grade. Therefore, the waiver test nicely defined the most important terms, definitions and concepts, as decided by the committee. Although each instructor might identify other content items that might be important, in effect, the list of

items on the waiver test constituted the minimum mandatory content items that the students must learn in this course. Some sample exam items were:

> Computers that are more powerful than microcomputers and can support a number of users performing different tasks are called
> A. Personal computers
> B. Small computers
> C. Supercomputers
> D. Minicomputers
>
> What general application software allows a user to perform financial analysis such as making budgets or projects?
> A. Word processing software
> B. File and database management software
> C. Graphics software
> D. Electronic spreadsheet software

As an instructor I decided to accept this list of items and focus on how to best teach these to my students so that they would achieve the best score possible on the comprehensive final exam.

I came to believe that a certain amount of simple familiarization, if not outright memorization, would assist my students in learning these basic content items and that repetition would have major advantages. However, I felt that limited class time restricted the amount of repetition I could present in the classroom. I started to consider the possibility of creating a computer program might be able to present these items to the students in the format they would view them on the final exam, multiple-choice questions, and provide them with immediate feedback and repetition, if necessary. I believed that such a computer program, which the students could use outside of class, could be extremely beneficial to them. I started discussing this possibility with my colleagues and experimented with writing such a program.

Actual Development of the Computer Instructional Aid

As described in chapter 3, my intent was to follow the typical Systems Development Life Cycle (SDLC) recommended for creating a computer program for business. This is a five-step development process: analyze the problem, design the program, create (code and test) the program, implement the program, and maintain the program. These steps are comparable to the steps recommended for the development of instructional materials described by Walter Dick and Lou Carey (1996). (Figures 3.3 and 3.5)

Analysis

During the 1996 and 1997 school year, I considered the requirements for a computer program that would be a computer instructional aid to assist students in studying for the common final exam. Normally, it is important to involve future users of a computer program in the requirements and design phases of the development of a computer program. However, since I have personally taught this course many times, I was familiar with the objectives and content of the course. As mentioned in chapter 3, I also had several informal individual conversations with Lori on the Owosso campus and Joyce of the Port Huron campus. Both were full-time instructors who had taught this course several times. They were both also on the committee that rewrote the waiver test the last time it was revised.

In these conversations I described the requirements in terms of my ideas, goals, and intentions regarding the development of the computer instructional aid. I then asked for comments, feedback, and concerns. If the instructor expressed a concern, I described

various alternative operations the computer instructional aid could be programmed to perform, and ask for her opinion. For example, Joyce was concerned with when the students would receive feedback. I explained that I could program the computer to provide feedback after each question, at the end of each chapter or when the student ended the program. I said I felt immediate feedback after each question was probably best. She agreed and stated that immediate feedback would have been her choice. Lori was concerned that the feedback be nonjudgmental. I explained that I could program the computer with any message we felt appropriate but that I was planning on a simple "Correct" or "Incorrect" as I had used in my previous programs. She replied that that sounded like a good idea.

Occasionally they made suggestions I had not considered. Joyce mentioned that even if a student quit the program before ending a chapter, it would be nice to provide the student with scores on those questions attempted. Since those scores would still be collected, it was a simple matter to add the displaying of the scores to the steps the computer would perform if the student chooses to quit.

But both instructors thought my requirements were important. All three of us agreed that the major concerns the computer instructional aid (computer program) should address were as follows.

1. Questions Should Be Similar in Appearance and Content to Questions on the Common Final.

The official course outcomes are attached in Appendix D. Generally, this course is intended to ensure all students have a basic understanding of computers and software.

If a student has had prior computer experience or education, they may take a waiver exam. If they pass the waiver exam with a score of 80% or better, they are not required to take this course. The waiver exam is the currently used as a common final for all INF111 courses on all campuses in the Baker College System. Some typical questions were:

The on-screen dots that can be illuminated are referred to as _____.
A. Monitor
B. Bits
C. Pixels
D. Icons

_____ is the process of making changes to the content of the document.
A. Creating
B. Editing
C. Formatting
D. Printing

The major purpose of the computer instructional aid was to help the student prepare for the common final used throughout the college system. The existence of a standard final exam helped clearly define the specific content that needed to be included in the aid. We needed a computer program that would provide the student with practice answering multiple choice questions as similar as possible to those on the final exam and included the material that would be tested on the final exam.

I had some concern that other instructors would not want use the actual questions from the waiver test in the computer instructional aid test bank. The waiver test is kept fairly confidential, especially since it is used as a common final exam on all campuses in the Baker College System. I used an encoding technique available with GWBASIC that prevented the user from directly examining the program and test bank. The user could only examine the questions one at a time as presented by the program. The user was also unable to print the questions in the test bank. At one point I even started rewriting each of

the waiver test questions in the test bank so the wording was somewhat different. However, all of the instructors I talked to were unconcerned about this matter. They felt that with a waiver test of 81 multiple-choice questions, it would be easier for a student to learn the terms on the test rather than just memorizing the answers to the questions (i.e. The answer to question number 16 is B.). So I included the actual questions from the waiver test in the computerized test bank.

2. The Program Needed to Provide Immediate Feedback on Correctness.

The major purpose of the computer instructional aid was to provide the student practice in answering multiple choice questions similar, or identical, to questions on the common final exam. Based on my experience with instructional design, I felt that feedback should be as instantaneous as possible. If the student has answered a question incorrectly, the computer instructional aid should inform the student as quickly as possible. If the aid delays a response until some later time, such as after a groups of questions has been answered, during the intervening period, the student has been allowed to assume that his or her answer was correct. This can be extremely confusing when the answer was a guess. Therefore, I felt it was important that the program respond immediately to a student's answer.

I also felt that providing immediate feedback would make the program more useful to the students. By identifying correct and incorrect answers, the student would quickly know which terms, definitions and topics required more study and which he or she already understood.

Immediate feedback is a common advantage of CAI programs and was relatively easy to implement. However, I had greater difficulty deciding what the program should do next. I discussed this with Lori and Joyce. In a "pre" instructor interview in March,1998 for the pilot study, Lori said the questions should be repeated if incorrect but "if right, they should not be asked again." She felt that feedback should be given immediately after each question. I finally decided that the student should be required to answer the question correctly, before continuing. After the pilot testing of the computer instructional aid in October, 1998 Joyce said, "I like the way that you give them the question and then, if they get it wrong, they get a chance to answer it again, immediately." The major alternative would be for the aid to provide the student with the correct answer if the student answered incorrectly. I felt it would be better if the aid required the active response from the student of providing the correct answer than the passive response of accepting the correct answer from the aid. Therefore, rather than giving the student the correct answer when a question was missed, the aid would require the student to look up the answer or at least guess until answered correctly.

3. The Program Needed to be Easy to Use.

Since this was a beginning computer course, no prior computer knowledge could be assumed. That meant the program must be easy to start and student choices must be few and simple. The program should also be easy for the instructor to use. The less instructor time and instructor preparation, the better.

4. The Program Needed to be Completely Student Controlled.

Results had to be confidential and there had to be no penalty for wrong answers. This would encourage the student to use the program. The student needed to be able to use the program in many different locations. By making the program independent of any specific computer or network, it would be easier for students to use.

5. The Program Needed to Appear to be Something the Student Would Find Useful.

I needed to title and organize the computer instructional aid so that the instructors and students would be inclined to believe this might be something that would benefit the students. Since both the instructors and the students were aware of the importance of the common final exam, it was important that the computer instructional aid be closely identified with that exam. And since the textbook was one of the major instructional aids being used by the teacher, it would also be beneficial for the aid to relate to the textbook.

Design

I decided to divide the question bank by subject areas that directly related to chapters in the textbook for this course. This permitted students to work on only one topic at a time. Working through the entire test bank would have taken several hours. By allowing the student to choose only one topic or chapter at a time, the student could work with all of the questions on that particular topic in a one sitting at the computer of less than an hour. Additionally, by dividing the question bank by chapter, the computer instructional aid could be used to review for individual chapter tests.

On starting the program, the student would be presented with a menu listing all of the chapters and allowed to choose the chapter to be practice (see Figure 4.1). Within each chapter the questions were to be randomly presented to the student to discourage the student from memorizing the answers by the order of the questions. The questions were to be multiple choice questions similar, often identical, to the questions on the final exam.

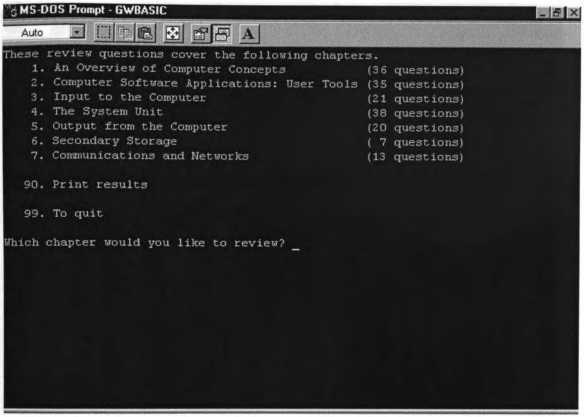

Figure 4.1. Menu of Computer Instructional Aid

The student would be presented with a multiple choice question on the computer screen and allowed to choose one of the answers (see Figure 4.2) by entering A, B, C or D. The program would also allow the student to enter a Q at this time if the student wished to quit working on that chapter. The program would immediately compare the student's answer with the correct answer as indicated in the test bank and inform the student if the choice was correct or incorrect. If correct, the student would be presented with another question from that chapter. If incorrect, the student would try again until he or she got the correct answer. This would allow the student to look up the answer in the textbook or just guess. In either case, the program wouldn't give the correct answer, just tell the student if the guess was correct or incorrect.

Figure 4.2. Examples of Questions and Feedback

After all questions in the chapter had been presented to the student, questions missed on the first try would be presented again. This would continue until all questions from that chapter have been answered correctly on the first try.

At any time, students would have the option of quitting the questions and returning to the menu. From the menu, the student could quit the entire program, try the same chapter again, or try another chapter. After a chapter was ended by either quitting or answering all of the questions correctly, the student had the option of printing out the results. These results include the date, time, chapter attempted, whether the chapter was completed or not, and the number of tries it took to answer all questions correctly (see Figure 4.3).

Figure 4.3. Example of Computer Instructional Aid Results

I also decided to encrypt or code the program, including the test bank so students couldn't alter it nor look at it. This was to prevent students from "cheating" and printing out the answers in the program and bypass the practice feature of the program.

Creating the Program

I wrote the program in GW-BASIC. While this does not allow a Graphical User Interface and the use of a mouse, it does permit the program to be used on any IBM PC-type computer, no matter how old. GW-BASIC also had the advantage that it uses an interpreter rather than a compiler. A programming language compiler converts the entire computer program into machine language by creating a second, machine language version, of the computer program. It is this second version that is the user than runs or executes. A programming language interpreter converts and executes each individual line of the computer program each time the program is run. This allows the programmer to run the program immediately after any change or addition to the program, a major advantage when using the prototyping approach. The program itself, including just a portion of the test bank, is attached (Appendix E). The basic program was written between January 12, 1998 and January 24, 1998. During this time fourteen different versions were created and saved. Additional changes and versions were made on June 17, 1998 and again on September 10 and 18, 1998. This took a total of about 60 or 70 hours and seventeen tries or versions of the program.

Using the experience gained in writing the earlier "Pilot-like" program written in 1988, I used a "prototyping" technique to create this program. That is, the central part of the program was created first, saved and tested. Then I would add and edit lines, resave the program under a new name (TRY1, TRY2, TRY3, etc.), and test it again. For example, I wrote the first try on January 14, 1998 to verify the major steps the program needed to perform and that those steps would successfully result in a program that met

the requirements. This first try, done on January 14, only included two questions in the test bank. On this first try the program:

(1) read the questions from internal DATA statements into an array or list,
(2) then a simple loop
    - read and displayed lines from the test bank as placed in the array or list,
    - identified and extracted the correct answer (indicated by an asterisk),
(3) when a blank line was encountered, the loop temporarily paused and
    - accepted the user's answer,
    - compared it with the correct answer from the test bank, and gave a "correct" or "incorrect" response to the user,
(4) then continued reading and displaying lines for the next question.

Once this portion of the program was working properly, additional elements were added and tested. For example, the second try added a variable for the student's name, counters for the number of questions correct and incorrect, and a printout of the results.

During this creation process the prototyping method allowed for a fair amount of experimentation. For example, I next needed to figure out how to incorporate a large test bank into the program. I wrote the third and fourth tries to see if I could write a program that could take a test bank created in a word processing program and convert it to DATA statements that could be merged into the existing program. I started by writing the TRY3 program that converted a text file of questions, such as would be created by a word processing program, into an ASCII file with a carriage return (CR) at the end of each line. In the TRY4 program I added line numbers, the key word DATA and quotation marks before and after the text on each line. By carefully choosing line numbers that did not conflict with existing line numbers, I was then able to perform a MERGE command in GWBASIC that added the DATA lines after the instructions in the program. These first four tries were all done on January 12-14, 1998 and took about ten hours.

I was next able to work on the program on January 17, 1998. I started by modifying the main program. In TRY5, I increased the number of questions to twenty-one and used the conversion programs mentioned above to add them. I also changed the names of some variables used to control the processing and simplified the subroutine to find the correct answer.

In TRY6, I added a tilde (~) character at the beginning of each question so that the program would stop displaying text lines and accept the user's answer. In earlier versions, I used the blank line between questions. This new technique provided more active control and avoided errors caused by missing or duplicate blank lines in the test bank. It also meant I had to add a special FIRST$ variable to prevent the program from requesting the user's answer before the first question.

I then started adding the lines needed to randomly present the questions. In earlier attempts, questions were simply presented in the order they were placed in the test bank. For TRY7, I added lines that found the number of questions in the test bank, then added a second array or list that contained all of the question numbers in random order. I completed adding the randomization factor on January 18 by creating TRY8. This version read each question number from that second array, then found and displayed that question from the first array.

The program was starting to get very complicated at this point. There were quite a few loops and branching decisions (IF statements). In order to reorganize the program, I increased the internal comment lines and reversed one IF condition. That meant I also had to move the two branching routines it controlled. I then resaved the program as TRY 9.

My next step was to renumber the program lines. Normally BASIC programmers increment line numbers by ten in order to make it easy to insert additional lines by creating a line with a number between existing lines. By this time I had inserted and moved so many lines that it was impossible to insert additional lines in some places. Therefore, I ran a renumbering operation to change all line numbers so they would be incremented by ten again. I saved this version as TRY9B because it only involved renumbering lines. Unfortunately my reorganization to create TRY9 resulted several incorrect references to line numbers in the program. Some references were to lines that no longer existed. My renumbering complicated these errors and required extensive debugging to identify and correct them. By the time this was completed, I decided to treat TRY9B as TRY10.

On January 19 I removed the lines identifying and checking for the FIRST$ question in the test bank. I no longer needed these lines since I had made the changes to read the questions in random order and reorganized the program. I also removed entries that were checking for blank lines and were no longer needed. I saved this version as TRY11. So tries number 5 to 11 were done from January 17-19 and took about fifteen hours.

With TRY12, I started adding the lines necessary to identify questions that were answered correctly so that the program could repeat only those questions that were answered incorrectly. I started by adding a second column or "dimension" to the array containing the random question numbers. In the version of BASIC I was working with, all variables are initialized at zero. If the question was answered correctly, the second value for that question was changed to a "1". This change meant I also had to change the

test condition to control the end of the program. I added a new variable, FIN$, which was set to "YES" at the start of each pass through the array of random question numbers. Each question was then examined to see if it had been previously answered correctly (second array value changed to "1"). If a question was found that had not been answered correctly, the FIN$ variable was set to "NO" and that question was presented to the student. If the FIN$ variable still contained a value of "YES" when the end of the array was reached, then all questions had been answered correctly and the program would end. If the FIN$ variable contained a "NO" at the end of the array, it was reset to "YES" and the questions were reexamined starting at the beginning of the array.

Since some questions would be answered more than once, the number correct and incorrect were not meaningful anymore. So I changed the variables to count the number of questions attempted or tried and the number of questions answered correctly the first time they were presented. Starting with this version the scores were displayed on the screen in addition to being printed on the printer. TRY12 try was completed on January 23.

On January 24, starting with TRY13, I started dealing with the changes necessary to allow the students to work with more than one test bank. This meant allowing the students to exit the program or restart it after all questions for a particular chapter had been answered correctly. I then added a program line that would allow the student the opportunity to exit the chapter whenever a question was displayed. I also added statements that would print the chapter number, date and time when the program printed the number of tries and the number of questions answered correctly on the first attempt.

In TRY14 I added a menu which allowed the student to select a chapter. The test bank for that chapter was saved in a separate program that simply read the questions into an array. Since each chapter test bank was part of a separate program, it could be saved in an encoded format that the students would not be able to read or access directly. This separate program was then linked to the main program. The main program randomized the questions for that chapter, presented them to the student, told the student if the answer was correct or incorrect, repeated the questions until all had been answered correctly or until the student indicated he or she wish to quit, displayed and printed out the number of questions attempted and the number of questions answered correctly on the first try, and returned the student to the menu. From the menu, the student could select the same chapter again, select a new chapter, or indicate if he or she wished to end the program.

At this point, on January 24, the program was essentially completed and ran correctly. I called this my "beta" version. As each feature was added and tested, bugs (errors) were relatively easy to find since they had to be in the new lines of code that had been added. Unfortunately, the complexity of the program also grew with loops inside loops, subroutines and branching all over the place. Even though I have had many years of programming experience, I found it difficult to maintain an overview of the entire program. The looping logic was fairly complicated and caused most of the problems in the final 6 tries. Even then, there were minor errors that were subsequently found. Tries number 12 to 14 were done on for 1/20 to 1/24 and took about 10 hours making a total of 35 hours to this point. I was ready to "pilot" the program with a few students in the winter and spring quarters of 1998.

Creating Instruction Sheet for Students

At the end of January, 1998, after testing the program several times myself, I wrote up a short set of instructions (Appendix F) and gave a copy to Lori on the Owosso campus. She had some problems getting the program to work properly until I realized I had not included the instruction to "Press caps lock button before running program." This step was quickly added. She tried the program twice and then made some suggestions. These included adding instructions for students on loading and running the program from a network drive as an alternative to the floppy disk. These drives were only accessible in Baker computer rooms but this way students could run the program even if there was a problem with the floppy disk. I also added commands to restart the program in case of an error message or if the program locked up (Appendix G). Lori also found a couple of errors in the test bank that I corrected.

Before the start of the spring 1998 quarter, on April 2, I made several changes to the instruction sheet. At the top, I added a paragraph describing the final exam and the review program. I also organized the instructions into seven numbered major steps.

On September 30, 1998, before the beginning of my full implementation of the computer instructional aid, I made some more changes to the instruction sheet. I removed the instructions for running the program from a network drive because that method would not save student results plus the program seemed fairly reliable at this stage. I edited the instructions for dealing with errors and problems in an attempt to simplify them. I also added my name, office hours and phone number in case students had difficulty with the program. (Appendix H)

In November 13, 1998 I finally modified the program so it would accept answers in either upper or lower case and removed that instruction from the instruction sheet. This last change was made just before I distributed it to the last class, Cynthia's in Flint.

Pilot Testing of Program

In the winter and spring quarters of 1998, two instructors gave the program to students in their INF111 classes. My major goal was to "field test" the program to find any errors or bugs. One of the instructors was Lori on the Owosso campus, the other was Joyce on the Port Huron campus.

Pilot Case #1: Lori on the Owosso campus

During the last week of winter quarter, about March 19, 1998, I interviewed Lori. As I mentioned before we shared an office at that time. We also shared materials and many opinions about this and other computer courses so actually we had discussed many of these topics before. I asked her two sets of questions: the first related to the INF111 course (see Table 4.1 or Appendix I) and the second related to the drill-and-practice computer program that I had written (see Table 4.2 and Appendix J).

Table 4.1
Pre-Instructor Interview - Course - Version 1

1. What do you think should be the purpose of the INF111 Information Systems Theory course?
2. What do you think should be the content of INF111?
3. In what ways can we give this content to students so that they can pass the common final for INF111?
4. How much lecture time do you usually spend in this class? How much do you think is really useful?
5. Do you think a "drill and practice" computer program using a test bank could help the students pass the final?

6. Do you think such a computer program could reduce the amount of lecture time? How much?
7. After introduction, should such a program be used during class time? Individually? Available on separate disks?

In answer to the first question related to the course, Lori stated that she felt the purpose of the course was an "introduction to computers for students with little or no experience." She went on to state that it covered "basic computers, printers and various software" plus the process of purchasing a computer and some coverage of the Internet. I skipped the second question relating to content since she had answered it in her reply to the question on the purpose of the course. The third question related to presenting the content that students needed to pass the final. Lori explained that she always included a review for the final. In reply to the question on lecture time, she stated that of the four hours a week the class met, she spent approximately two and a half to three hours in lecture but that only about one to one and a half seemed really useful. She felt that a drill-and-practice computer program would definitely help students pass the final and she hoped it might reduce the lecture time to the one to one and a half hours that seemed useful. In response to question seven, Lori said that the more ways the program could be made available to students, the better.

Table 4.2
Pre-Instructor Interview - Computer Instructional Aid - Version 1

1. What choices should be available to the student from the main menu? Choice of chapter? Multiple chapters? Cumulative (all chapters up to and including current one)? Complete (entire textbook)?
2. How should questions be presented? Multiple choice format? Random order? Random choices (A to D)? Repeat questions (always, only when incorrect)? Feedback?
3. What results should be provided? What printouts? What should be recorded (score, time)?

4. What should the instructor require of the student? How many and how often should printouts be required? What score (%) should be required of the student? Should the program require a specific score before allowing the student to continue to the next chapter?

In answer to question one related to the drill-and-practice computer program, she stated that however I divided up the questions for the main menu would be fine. Relative to question format, question two, we discussed the various options: true and false, multiple choice, short answer (fill-in-the-blank) and short essay. I explained that short answer and short essay would be difficult, it not impossible, for the computer to grade and she said that multiple choice matched the common final and should be fine. She said the questions should be repeated if incorrect but "if right, they should not be asked again." She felt that feedback should be immediately after each question. Relative to printouts, she said that some form of study guide or chapter summary would be nice. She also felt that it would be nice if the score and time could be recorded and printed but that mainly, she needed some sort of printout that certified the student had completed the chapter. She also wanted printouts for each chapter and liked the idea of presenting questions answered incorrectly until ultimately 100% of the chapter questions had been answered correctly. She did not believe the program should require a specific score before allowing the student to continue to the next chapter.

During the summer of 1998, Lori again used the program in her INF111 classes. There are four basic INF courses that almost every student has to take: INF111, Information Systems Theory; INF112, Word Processing; INF113, Electronic Spreadsheets; and INF114, File Management. Few students choose to take classes in the summer so there are limited offerings. There were only two sections of each, one in the morning and one in the evening. In order to accommodate Lori's personal schedule, I

agreed to teach all of the Monday-Wednesday courses and she taught all of the Tuesday-Thursday courses. This meant that we each taught one of all of the courses. We ended up sharing a lot of our materials and one evening I substituted for Lori when she had a conflict. In other words, we worked fairly closely on teaching these courses.

During this early testing of the program all necessary files were placed on a network drive and each instructor helped her students copy the program to a student disk. The instructor then had the students load and run the program at least once in class to be sure the students knew how to use the program. Since limited sections were being offered, these summer classes were all taught in a computer room, which simplified this process.

During the summer quarter of 1998 I distributed a consent form and survey form to Lori's class, my summer class, and Joyce's summer class (see next case). At that time, the student survey form consisted of four questions related to the computer instructional aid added to the end of 14 questions that were used on the standard course review used by the Baker College System to obtain input from students to help evaluate a course. All questions involved a statement followed by a Likkert scale from one (strongly disagree) to five (strongly agree) plus a not applicable choice (NA). The four statements related to the computer instructional aid were located on the back of the form. At the end of the class, on July 20, we received signed consent forms and survey forms from four of her students. The responses from the other students are shown in Table 4.3. After each statement, space was provided after a "COMMENTS:" heading. After the last question once student wrote, "It made the class much easier for me. It also helped me a great deal in getting more comfortable with my computer at home."

Table 4.3
Student Survey Questions – Pilot Case #1

STUDENT SURVEY QUESTIONS - PILOT	Strongly Disagree				Strongly Agree	
	1	2	3	4	5	NA
15. The review program is a useful learning resource.				1	3	
16. The review program is very helpful in preparing for the final.				1	3	
17. The review program was simple and easy to use.				1	3	
	0-2	2-4	4-6	6-9	> 9	NA
18. I spent approximately the following number of hours using the review program.	1		2	1		

Table 4.4
Post-Instructor Interview – Version 1

1. Did you like using the computerized questions and answers?
2. Do you believe the computer program was a positive or a negative part of the course?
3. Any suggestions to make the program better?
4. Is there another review method that you think is better than the questions of the computer?
5. What problems did you encounter with the program?
6. What features of the program did you like?
7. Would you like to use the program the next time you teach this course?

I also interview interviewed Lori on August 27, 1998 at the end of the summer quarter (see Table 4.4 or Appendix K). When asked if she liked the computer instructional aid, Lori replied that "Yes, it was good preparation for the common final." "The quantity of questions was great, the students did better and scored better." "Several students ran it multiple times." The students "used it in class and at home." She felt the aid was a positive part of the course. She gave credit for doing it and used class time for it. Some suggestions she thought might help make the program better included an option that allowed the student to print a statement that "chapter x completed at 100%," the

ability t

it into s

there w

questio

no prob

requirer

to the fa

the writ

would b

that stu

asked it

notice."

the cou

elsewh

1998 w

the pro

bugs a

break b

and wa

I starte

ability to stop in the middle of a chapter, reducing the length of each chapter by dividing it into sections, and a "hard copy" printout of the questions. She said that she did not feel there was a better review method. That practice was best although a hard copy of the test questions that had to be returned at final exam time might help. She said that there were no problems encountered with the program then remembered that the "caps lock button requirement did cause some problems, many students didn't catch that." She was referring to the fact that the program required the caps lock button to be pressed. This was stated in the written instructions for the program but if a student missed the instruction, all answers would be treated as incorrect. She said she liked the "entire program," especially the fact that students often "looked up the answers" in the textbook if they were uncertain. When asked if students seemed to be guessing at some of the questions she said she "didn't notice." She stated she would definitely like to use the program the next time she taught the course. Unfortunately, shortly after the interview Lori left the college to teach elsewhere and therefore did not participate in the full study in the fall of 1998.

Pilot Case #2: Joyce on the Port Huron campus

The other instructor who introduced the program to her students prior to the fall of 1998 was Joyce on the Port Huron campus. Joyce had also offered the "beta version" of the program to INF111 students in the winter quarter and spring quarter to help identify bugs and problems in the program. I met with Joyce in March of 1998 during spring break between winter and spring quarters. I had started developing interview questions and wanted to try out my questions and get some substantial input from other instructors. I started with the first set of interview questions that were related to the course (Table 4.1

or Appendix I) The first two questions had to do with the instructor's perceptions of the purpose and content of the INF111 course. Joyce admitted that she wasn't sure what the purpose should be but primarily viewed the course as providing (1) a background for more advanced computer courses and (2) a basic understanding of "how a computer works." As to content, she supported the "traditional" emphasis on "terms and terminology" but also felt it was important that students "understand the process" the computer uses to work with data.

In response to the third question, "In what ways can we give this content to students so that they can pass the common final for INF111?" Joyce stated that she felt the common final "needs to be changed." She explained that she used study sheets to help the students prepare for the final and that is was important to "match the course content to the final." The fourth question related to amount of lecture time. Joyce said she spent about 2 of the 4 hours each week in lecture, discussion, and question and answers but only one hour of lecture was really useful. To question five, "Do you think a 'drill and practice' computer program using a test bank could help the students pass the final?" Joyce simply responded "sure." In response to the question, "Do you think such a computer program could reduce the amount of lecture time?" she said, "Yes, especially when compared with traditional methods." She then responded with a "yes" to each part of question seven: "After introduction, should such a program be used during class time? Individually? Available on separate disks?"

I then proceeded to the second set of questions that were related to the computer instructional aid (Table 4.2 or Appendix J). She said that yes, students should have a choice of chapters. She thought if the student could choose multiple chapters and have

the computer randomly choose questions from those chapters that would be nice. It would also be nice if the program could randomly present questions from all chapters up to and including the current one or from the entire textbook. She thought that most important options would be the ability to choose questions by chapter and by test (which might cover two or more chapter). She thought the multiple-choice format was good but she didn't like true and false questions. She found the possibility of short answer questions interesting but felt they would be "much more difficult" for the computer to grade. Yes, she thought it was important that questions be presented in random order. She admitted that she had never thought of randomly mixing up the choices for a given multiple-choice question and didn't have an opinion on it one way or the other. She thought it was a "nice idea" to have questions answered incorrectly thrown back into the test bank but also said "I'm not sure I like it to keep asking the same question. It's like beating a dead horse." In response to the question about printouts she said, "I like printouts. If it's nothing more than their name and score type thing that they can only get from finishing it. So that as an instructor they can bring me a piece of paper and say 'See I did it.' So that I can give them credit or whatever. That's how I do some of the hands-on exercises. I mean, I don't really care how well they did, it's did they bother to do it." She then made it clear that she was not concerned with giving the students the test bank questions, which included the actual questions on the final exam. She said "I do that with the review sheet" that she gives them to study for the final. We then discussed the problems with arriving at a score if questions answered incorrectly are presented to the student a second or even third time. She thought it might be useful to know "how many questions did they really answer. Which would tell

you how many they were redoing." In discussing whether the time should be recorded, she mentioned the review sheets again.

> "Traditionally what I do with the review sheets is, here's the review sheets. You want to use them, fine. You don't want to use them, that's fine too. It's up to you. This is what's going to be on the test. It's up to you what to do with them. And dump the responsibility back on to the student. And then I don't have to take class time to worry about it cause it's their responsibility. Class time can be spent doing more fun things like surfing the web or exploring other options. The problem is coming up with options that aren't too hard for them and figuring how you can get 30 people into a 24 computer lab. I don't know. If I was going to require it, if I was going to spend class time doing it, then I would say they need to give me printouts once a week."

Joyce also gave the program to students during the summer of 1998. By that time I hoped to finalize the program and interview process. I made some minor changes to the program on June 17th and felt it was ready for final testing. Since I had personally interviewed Joyce in detail in March, I emailed her my new list of interview questions for instructors before the course begins (Appendix L) and she emailed answers back to me on June 22, 1998. Her answers were fairly short and to the point. Yes she thought the program might be helpful and we could make arrangements to show her students how to use it. She thought we could distribute the program by "either giving them a disk with the program on it or allowing them to copy it off of the system." By system, she was referring to the network system, LAN, which allowed instructors to place things on the server drives so students could copy them to any computer on the network. Yes, the program should allow students to try again when they miss a question. Yes, students should be able to print out questions, "but maybe not the answers." The program should print out "how many questions the student got correct and how many tries it took to get

them correct." She couldn't think of anything else the program should do nor any possible problems or concerns.

The summer quarter began on June 22. After making several changes to the program on June 28 and 29, I emailed the program, as an attachment, to her. At the end of the class, on July 21, we received signed consent forms and survey forms from 9 of her 13 students. One student who signed a consent form and filled out a survey form did not answer any of four statements on the back of the form related to the computer instructional aid. The responses from the other students are shown in Table 4.5. After statement number 15, one of the students who marked five wrote the comment, "It's a good study tool!" After statement number 17, a student who marked three wrote, "I think it needs a little more fine tuning." After the last question one student wrote a general comment about the instructor and the course. The student who marked low values for all of the last four statements wrote the comment, "This class was very helpful and useful especially as a review for myself."

Table 4.5
Student Survey Questions– Pilot Case #2

STUDENT SURVEY QUESTIONS - PILOT	Strongly Disagree				Strongly Agree	
	1	2	3	4	5	NA
15. The review program is a useful learning resource.		1		2	5	
16. The review program is very helpful in preparing for the final.	1			1	6	
17. The review program was simple and easy to use.		1	1	1	5	
	0-2	2-4	4-6	6-9	> 9	NA
18. I spent approximately the following number of hours using the review program.	1	4	3			

On August 30th I interviewed Joyce about her experiences using the computer instructional aid using the second version of questions for instructors at the end of the

course (Appendix M). She said yes, she like using the computerized questions and answers and it was a positive part of the course. It required "less review time" in class. She thought it would be better if it could be run from Windows. She said some students preferred review sheets. There were some minor problems getting into the program and some difficulty in returning to earlier chapters in the menu. She especially liked the feature that allowed the students to try again. and yes, she would like to use the program next time.

Pilot Case #3: Jim on the Owosso campus

I also taught a small section of INF111 in the summer of 1998 and gave the program to my students. I had five students fill out consent forms and survey forms (Table 4.6). I had two students who wrote comments after almost every one of the last four questions. After question number 15 regarding usefulness, one student wrote "Worded extremely well for me! Very helpful." Another wrote "It was fun to use." After the question on helpfulness, the first wrote "very helpful" and the second wrote "definitely a big advantage." After question 17 on ease of use, the second student wrote "I wish every class had one." Finally, after the question on number of hours using the program, the first wrote "Very easy to use, worked well with me, was a very good study tool to have. Loved it!" The second student wrote, "It only takes a few minutes to go through each review. Excellent."

Table 4.6
Student Survey Questions – Pilot Case #3

STUDENT SURVEY QUESTIONS - PILOT	Strongly Disagree				Strongly Agree	
	1	2	3	4	5	NA
15. The review program is a useful learning resource.					5	
16. The review program is very helpful in preparing for the final.					5	
17. The review program was simple and easy to use.					5	
	0-2	2-4	4-6	6-9	> 9	NA
18. I spent approximately the following number of hours using the review program.	3		1			1

Summary of Pilot Testing

The totals for all three groups, 17 students, is as follows:

Table 4.7
Student Survey Questions – All Pilot Cases

STUDENT SURVEY QUESTIONS - PILOT	Strongly Disagree				Strongly Agree	
	1	2	3	4	5	NA
15. The review program is a useful learning resource.		1		3	13	
16. The review program is very helpful in preparing for the final.	1			2	14	
17. The review program was simple and easy to use.		1	1	2	13	
	0-2	2-4	4-6	6-9	> 9	NA
18. I spent approximately the following number of hours using the review program.	5	4	6	1		1

A large number of students who filled out an evaluation form found the computer instructional aid useful, helpful and easy to use. I feel justified in trying to distribute this aid to more instructors and students. I also wanted to measure the results more accurately. So I decided to continue working towards wider distribution of the program.

Changes to Program

On June 17, 1998 I removed the menu from the main program and put it in a separate sub-program as I had done with the test banks for each chapter. One of the problems with this version of the program was that it required eleven files on the student disk. This included the GWBASIC program itself, one file for the main program, one file for the main menu, and eight files or programs for each of the chapter test banks. I experimented with several alternative methods of saving the test bank. In prior tries, each test bank was saved within a separate program. GWBASIC allowed me to save each program in an encoded format so it could not be read but only executed. TRY15 involved saving all of the test banks as one large ASCII text file. Although this worked, I could not figure out a simple way of encoding the data to prevent students from examining it directly using a word processing program.

For the piloting of the computer instructional aid I wanted to distribute the program during the second week of the summer quarter. Therefore on Sunday, June 28th, and Monday, June 29th, I made several changes to the program. I put the menu and all of the questions in one sub-program in order to decrease the number of files that I had to distribute. I added several comment lines for internal documentation to clarify what the program was doing at different places and in different sections. I had problems with the menu not printing. I also had problems with array subscripts not being large enough because the end of data was not being detected. I made some changes to some looping controls to solve the problem. I also had problems with the program detecting the end of each chapter test bank since all questions were now combined into one large sub-

program. Finally, I combined the questions from chapters seven and eight since there was only one question from chapter eight.

After the piloting of the program in the summer of 1998 I made several more changes to the computer instructional aid before distribution of the computer program in the fall of 1998. On September 10 I combined the program and sub-program to make one large program. I found that the entire program was still less than 60 k (kilobytes) in size. So I decided there was not any real advantage to placing the chapter test banks in different program or files. This also meant that that the student's disk only needed to contain the GWBASIC program and the one large program I had written. On September 13th I moved the section to find a read a chapter's questions to a subroutine to better organize the program. I also added a subroutine to print student results to a file on the floppy disk and to printout a statement listing chapters completed. This last item was in response to Joyce's request during our March 1998 interview when she requested the program create a printout "so that as an instructor they can bring me a piece of paper and say 'See I did it.'" I also added additional internal comments that identified variables.

Dissemination, Implementation, and Testing

Implementing the Program

Once the computer instructional aid was complete and tested, I made copies of the instruction sheet to hand out with the program. I obtained approximately 200 3-1/2" floppy disks. I printed labels for the disks that said "INF111 REVIEW PROGRAM, Please return to: Jim Newtown, Baker College of Owosso." I hand-wrote a number in the corner of each label so I could uniquely identify each disk. I put GWBASIC and an encrypted version of the final INF111 review program on the disks and distributed the program to six classes in the fall of 1999.

I approached four different instructors on four different campuses with the possibility of offering the program to their students. One of those instructors, Joyce on the Port Huron campus (Pilot Case #2 and Implementation Case #1), had been consulted during the design and creation of the program and involved in the piloting of the computer instructional aid. I was vaguely familiar with Judy on the Owosso campus (Implementation Case #2) where I teach. I knew she had taught the INF111 class before and was asked to take Lori's sections. The dean on the Muskegon campus responsible for the INF courses suggested the instructor, Tina (Implementation Case #3), to me. The fourth instructor, Cynthia on the Flint campus, (Implementation Case #4) was primarily chosen because she taught one of the few sections offered the second five weeks of the quarter rather than the first five weeks. This simplified scheduling visits to her class and made it easier for me to add one more section and one more campus to the study. I also verified that she had taught the class before.

In each situation, I interviewed the instructor before the beginning of the course (1) to enlist her assistance in the study, (2) to identify her prior attitudes about the potential usefulness of the computer instructional aid, and (3) to identify any possible last-minute modifications that could make the aid more useful to the instructors and students. I then visited each of the classes involved in the study; distributed the floppy disks, instructions and consent forms; and demonstrated the program. I returned to the classes when they took their exams, distributed survey forms, then collected the forms after the exam. I also then submitted follow-up interview questions to the instructors at the end of the class.

Implementation Case #1: Joyce on Port Huron campus

Prior to the start of the quarter, on September 21, 1998, I had made arrangements to introduce the program to Joyce's classes during the second week. Joyce had worked with me on the pilot and was familiar with the program and my intentions. We had decided to give the program to her 8:00 and 10:00 classes but not her evening 6:00 class so we could make some comparisons. She had agreed to require the program for her 8:00 by giving them credit for using it, but not her 10:00 class.

Joyce also planned on giving the common final the first day of class as a pretest. Joyce, and some other instructors including myself, have occasionally done this in the past. She told the students she wanted them to understand what material was covered in the class. When given as a waiver test anyone who passes with a score of 80% or better can waive the class. She did not tell students that the same test is used as a waiver test

and as the common final but did tell the students that, if they consented, those test scores would be made available to me.

On September 30th, Wednesday of the second week of classes, I went to the second hour of her 8:00 to 9:50 class and the first hour of her 10:00 to 11:50 class. These classes were held in a computer room and each student had a computer. I explained my project and the computer instructional aid. I then walked them through process of running the program and they each had the opportunity to use the program for 30 to 40 minutes. During this period we discovered that the program still had problems letting students go on to another chapter after they had finished one. Students had to exit the program and restart it in order to choose another chapter. I fixed this problem by the end of the week and emailed Joyce an updated copy of the computer instructional aid.

Table 4.8
Pre-Instructor Interview – Version 2

1. Do you believe this is an important class? Why or why not?
2. What do you feel should be the most important part of this class?
3. Are you satisfied that the common final appropriately tests what is important in this class?
4. What aids and tools do you use to help students pass the final exam?
5. Do you think a computer program that presents questions from a test bank might help your students review and prepare for the final?
6. How difficult is it to use a computer room during class time?
7. May I use some class time to distribute this program and demonstrate it to your students?
8. Would it be possible to require your students to turn in the printout indicating they used the computer program?
9. What questions would you like in the test bank used by the computer program?
10. Do you think the student should be able to print out the questions?
11. What results, if any, would you like the program to print out when the student ends the program?
12. Is there anything else you would like the program to do?
13. What problems and concerns do you have regarding the use of this computer program in your class?

I interviewed Joyce on Saturday, October 3, 1998. This was actually after I had already distributed the program to her class which was a little unusual. However since I had known Joyce for many years and she had worked with me on the pilot I knew her general enthusiasm for the computer instructional aid and her willingness to cooperate. Still, I was pleased for the opportunity to solicit her responses to some changed and new questions. The first four questions of the interview (see Table 4.8 or Appendix L) were aimed at identifying their opinions and attitudes regarding the course. I hoped to find a common ground with my perceptions of some of the problems with this course. I also hoped this would help me develop an atmosphere of mutual respect and communication in which they would feel free to express their opinions.

In answer to the first question, do you believe this is an important class, Joyce said "Until recently I was thinking it was a very important class, but as of late I'm beginning to feel like the students." She went on to describe the problems of dealing with a wide range of prior student knowledge and experience especially the boredom of those who already know most of what's covered in the class. "But on the other hand, is the terminology important? Yes." You have to talk so "people understand what you're saying. If you have to call everything thingamajigs or whatchamacallits and that blinking thing" then you are not going to be considered computer literate. In response to the second question regarding important parts of the class she repeated terminology. She added "but also to make sure that all those students that have never worked on a computer, used a computer, become a little comfortable with it and expose them to some different types of software." Joyce said she was "somewhat" satisfied with the common final. She then described chapter "review sheets that are fill-in-the-blank questions" that she had given to

students in the past and would continue to give to her class that would not receive the computer instructional aid. Those classes that received the computer instructional aid would only be receiving a list of terms prior to each chapter.

Questions 5 and 13 related to the instructor's acceptance of the computer instructional aid. Joyce "definitely" liked the program, especially since, in the pilot, "all students that commented on it at all were very favorable, and most commented." She did have several concerns. She was undecided as to whether she was "going to have to find ... class time to work on it or can I just require them to spend time outside of class?" From the pilot study she knew that most students "liked the program as opposed to the sheets because they liked seeing the actual questions, they liked the instant gratification, I got it right or I got it wrong." However, she was also aware that many students had a "lack of computer availability and time availability."

Questions 6, 7 and 8 related to the feasibility of integrating the computer instructional aid into the class. Joyce repeated what I already knew from prior conversations. The class was taught in a computer room so that wasn't a problem. There would be no problem with distributing and demonstrating the program during class time. She repeated our prior arrangements to require her 8:00 class to use it but not the 10:00 class.

Questions 9 to 12 related to instructor requirements of the computer instructional aid. She mentioned "it's nice to have a few extra ones" [questions] in the test bank that were not on the common final. In response to the question of students printing out the test she said, "truthfully, I don't care." In terms of printouts, she wanted to know "did they finish the chapter or not, how many did they get right, how many times did it

take them. Cause truthfully, I don't care how long it took them, or how many times they had to answer a question to get it right.... I really want to know did they finish or not finish it." But then added, "I think it would be cool to know how many times did they try it." In response to the question about anything else the program might do, she said:

> I like the way that you give them the question and then if they get it wrong they get a chance to answer it again, immediately. And then, at the end, [if] I understand it correctly, now that at the end, any question that they missed the first time through, they're asked it again. Before they finish the chapter. So even if it took them four times to answer the question or two times, if you missed it, you get it again. So it's reinforcing those they missed.

She concluded by saying "I like that aspect of it because I think repetition is one of the ways we learn."

On October 8th, Joyce sent me an email message about a problem one student was having with the computer instructional aid when she was working at home. It turned out I wrote the program to be used in drive A and on her home computer the 3-1/2" floppy disk drive was labeled B. On October 12th she sent me the following email message:

> My students found a problem in the program (I don't think its that big of a problem). The correct answer for the question 'What is a bay?' keeps changing. They said that one time the correct answer would be D and the next time A or B or C, my guess - Is a correct answer marked?

Upon investigation I found that indeed, in spite of all the testing and debugging, there was no correct answer indicated in the test bank for that question. Since the program did not identify a correct answer, it used the correct answer from the previous question. Since the questions were presented in random order, the answer to the previous question would usually be different each time. I was able to fix this for the two sections that I worked with in the last five weeks of the quarter.

Twenty-one students in Joyce's 8:00 class signed the consent forms, received the program and submitted a survey form at the end of the quarter (see Table 4.9). I only had pretest scores for 19 students since two students missed the first day of class.

Table 4.9
8:00 Port Huron Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Pretest	19	32	69	50.03	7.4685
Final Exam	21	67	100	89.00	10.2225

Student survey responses for the 8:00 Port Huron class to the first question, which related to overall perceptions of usefulness, are summarized in Table 4.10. Values represent number of responses to the survey question at that level. Immediately after question one, students were encouraged to enter any "Suggestions for making the program more helpful." One student wrote, "One of the questions (What is a bay) gives a different answer each time. Also make it so answers may vary different options because we can memorize the answers." Another student, who did mark "very useful" added "if you didn't have to end out of it after each chapter you completed!"

Table 4.10
8:00 Port Huron Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	18	2	3		

Responses to the second question are summarized in Table 4.11. Students were encouraged to "Please describe any problems you had." Nine students mentioned the problem of moving from one chapter's questions to another's. The program would give them a score without presenting the new questions. They had to exit and restart the program. Two students mentioned the problem with the question about a bay accepting different answers as correct each time. Two students said they were unable to get the

program to work at home, one student could not get the program to save the score for one of the chapters, and one student mentioned the program "froze on me a couple of times."

Table 4.11
8:00 Port Huron Class - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	10	8	2		

This third question was open-ended. Some students' answers fell into more than one category. Although this was the only class in the entire study that was specifically required to use the computer instructional aid, only five students mentioned that. Note that several students gave more than one answer and therefore were counted in more than one category. In categorizing answers I was interested in whether students focused on what encouraged them to use the program in the first place or on what caused them to continue to use the program after they had tried it. Therefore, if a student responded, "to help me study" I evaluated this as expectations motivating the student rather than experience with the program. If a student's comment started with "to study" I placed this in a separate category because I decided this was too unclear to put in the expectations or experience categories. Since both of the Port Huron classes were introduced to the computer instructional aid in class and used it for a half-hour, they could be expected to focus on why they continued to use it. That was definitely the case with the 8:00 class since 12 students said they used it because they found it helped and no students mentioned their expectations. Answers not summarized in Table 4.12 included, "I like questions," and "Only means of review for the class." One student also listed the instructor's name after "To study."

Table 4.12
8:00 Port Huron Class - Student Survey Question 3

STUDENT SURVEY QUESTION 3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? What influenced you?	5		12	5	1

Question 4 rated high answers (Table 4.13) but this was to be expected considering this was one of the four sections that received about an hour's worth of hands-on time when the program was distributed. Although comments were encouraged after this question, the few responses related to the program rather than the instruction. These included ""It was a great review," "Great studying device," "Very awesome!!" and "This program helped out a lot."

Table 4.13
8:00 Port Huron Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	19	2			

In question number 5, Baker lab referred to an open computer room at Baker College. LSS referred to the Learning Support Services, which is an open computer lab where tutors are available. Four students did not respond to this question nor any of the other questions of the back of the survey form. Students were encouraged to "Check all that apply" so many students checked more than one location.

Table 4.14
8:00 Port Huron Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	9	2		13		

Question 6 dealt with the chapters that were used in the computer instructional aid. As mentioned above, four students did not answer any of the questions on the back of the survey form. Answers are summarized in Table 4.15. The mean number of chapters reported covered by these students (6.5294) was a little higher than the mean of all classes (5.5775) but then this section was required to use the program.

Table 4.15
8:00 Port Huron Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
The software covered 7 chapters. Please check all you used:	1				1		15

I had hoped that the program would record both the number of chapters completed and the amount of time spent working with the program. Due to an error in my programming, it only captured the former. Even though the program recorded the number of chapters completed and question 6 on the survey asked how many chapters were used, there was an almost perfect match for this class. The only exception was the student who reported only one chapter used, did not complete that chapter according to the data saved by the program. Of the four students who did not answer question number 6, disks were returned by all four of them and indicated two had competed four chapters and two had completed all seven chapters. However, this data was not used in Table 4.16 since this data was not gathered as a response to question 7.

Question 7 asked, "Think back over all of your use of the software. How many hours total did you use the software?" This question was difficult to summarize since many students used qualifying adjectives (e.g. "about 2," "approximately 7 hours," "not sure possibly 4," "5 or more," etc.). For summary purposes, these were ignored. Another problem was that some students used fractions (e.g. "3 1/2" hours) and ranges (e.g. "3 -

4" hours). Unless specifically noted, these ranges were always one hour wide, as in "3 - 4" hours. To summarize for tabulation purposes (Table 4.16) a conservative approach was taken so fractions were dropped and the lower value of a range was taken. To summarize statistically (Table 4.17), fractions were converted into decimals and the midpoint of a range was used.

In addition to the four students who did not answer any questions on the back of the form, two other student responses were not included in Table 4.17. One was the indefinite response "hours." This was treated as if the student did not answer the question. One student did respond "about 6 hrs a week." The students were given the program on Wednesday of the second week and their final exam was on Wednesday of the fifth week. Six hours a week for three weeks gives a total of eighteen hours. This seemed unusually high since the largest number of hours reported by anyone else in the entire study was 10. I decided to include this student's response as a "no answer" since it was either a misunderstanding or clearly a statistical outlier.

Table 4.16
8:00 Port Huron Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?		1	3	4	1	3	2	1			

Table 4.17
8:00 Port Huron Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?	6	15	1.33	7	3.92	1.7482

Question 8 on the student survey form simply asked the student to "Please share with me any suggestions, criticisms, or advice that would help me improve the software." Fifteen students did not respond. One student referenced her answer to question 1 where she suggested randomizing the answers to each multiple choice question. Two of the other answers referred to the problems with the question related to a bay and the need to exit the program after each chapter. Four of the students, including one that mentioned the problems with the program, made extremely positive comments such as: "It was very helpful in reviewing for the tests! Thanks!"; "I think it was an excellent idea and I don't really have anything to say about it except it really helped me."; "I thought it was great!"; and "I really liked the program and was glad I got to use it."

Nineteen students in Joyce's 10:00 signed the consent forms, received the program and submitted a survey form at the end of the quarter (see Table 4.18). I only had pretest scores for 17 students since one student missed the first day of class. Two students did not take the pretest nor the final exam. This class was not required to use the program and received no credit for using it.

Table 4.18
10:00 Port Huron Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Pretest	17	27	73	49.74	12.6717
Final Exam	16	48	99	83.94	14.9330

Student survey responses for the 10:00 Port Huron class to the first question, which related to overall perceptions of usefulness, are summarized in Table 4.19. Values represent number of responses to the survey question at that level. Comments included: "Make it easier for the user to go to the next chapter without resigning in."; "Offer the great idea to all my teachers."; "Every class needs this."; "More time at the computer.,";

and "It's already very helpful just make it so you don't have to shut down to get to next chapter." The one student who indicated the software was a little bit useful wrote, "Useful because I could only use it in class but not useful because I don't have my own computer and limited access or time to use one."

Table 4.19
10:00 Port Huron Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	15		1		

In response to question 1, students generally found the software easy to use (Table 4.19). Comments included: "When choosing a new chapter I had to close and start the program over a couple of times."; "One answer was wrong all the time. The one about the bay."; "One question kept giving the wrong answer every time. It would change each time you answered it. Question was 'What is a bay?' Correct answer was D."; "I answered a question correctly or with every available answer and it still told me I was incorrect."; "If a person knows the procedure of the computer, this makes it easier."; and "Had to shut down every time for new chapter or same chapter review."

Table 4.20
10:00 Port Huron Class - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	13	3			

This class was not required to use the program so no student mentioned requirements or credit in answering open-ended question number three (Table 4.21). However one of the reasons one student gave was "teacher offered it." Another included the response "my teacher and fellow class mates." A third student was even more specific: "Seeing other students using it, I saw that it helped them to study for tests."

Students who indicated that they used it because they tried it and found it helped included statements such as "Much easier way to study," "It was so much better than reading a book. And you have to get the right answer to go on," and "It helped me immensely."

In spite of the fact that students worked with the computer instructional aid in class on the day it was distributed and saw exactly what the aid would and would not do, three students mentioned their expectations and the program's anticipated benefits. Remember that some students gave more than one answer to question 3 and were counted in more than one category.

I was unable to categorize several student responses. One student included the comment, "to test my knowledge of what I read out of the book." Another added the statement, "to learn the terminology because I just purchased a computer." Another, as his second reason, simply said, "the book, the program was easier." One student, after mentioning that he used it "to study for tests and final exam" added, "easy access."

Table 4.21
10:00 Port Huron Class - Student Survey Question 3

STUDENT SURVEY QUESTION 3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? What influenced you?	2	3	7	5	1

In response to question number 4 relating to adequate instruction, several students' comments related more to the computer instructional aid than the instruction in using it (Table 4.22). One student said, "This was the best way I ever saw for studying for a test. I did real well with this system. Thank you very much." Another student commented, "Without disk I don't think I would have retained the knowledge of all the terms." The student who responded "good" did not add a comment. The student who

marked "adequate" simply added "OK." The student who indicated instruction was "inadequate" provided a very detailed comment:

> I had trouble loading the program quickly enough to use in class or before class. Also once when I successfully loaded it, it would not allow me in, even using directions and commands on the handout. Needs fewer steps.

Table 4.22
10:00 Port Huron Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	13	1	1	1	

Question 5 was the first answer on the back of the form (Table 4.23). One student left the entire back blank but many students marked several places. The student who marked "other" indicated the location was the public "library."

Table 4.23
10:00 Port Huron Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	10	2	2	11		1

Most students indicated in question 6 that they covered all seven chapters (Table 4.24). There are only 15 responses because one student left the entire back of the form blank.

Table 4.24
10:00 Port Huron Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
The software covered 7 chapters. Please check all you used:		1	1				13

Again, summarizing hours for question number 7 was difficult because of qualifying descriptors such as "maybe" and "or more" so these qualifiers were ignored.

The same processes used for the 8:00 Port Huron class for summarizing data were followed for the 10:00 class. The one answer that was in the form of a range was three to four. For tabulating purposes (Table 4.25) this was converted to three hours, for statistical summary purposes (Table 4.26), this was converted to 3.5 hours. One student did add the comment, "When studying for two chapters at a time I studied for about a 1/2 hr. or 3/4 hr." Only one other student added a comment to question 7, "It is a very good program," which seemed to be more of an overall comment. This student did not comment to question 8.

Table 4.25
10:00 Port Huron Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?		1	3	4	1	1		1	1		3

Table 4.26
10:00 Port Huron Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?	1	15	1	10	4.90	3.2304

Eight students answered question 8's request for suggestions, criticisms or advice. Some of these were very short: "Perfect"; "I think it was wonderful"; "Make it easier to go from one chapter to another."; and "Make it available for all classes." One student wrote, "The program kept going over the questions. It should end when done with the chapter and then ask you which chapter you want to review." Another said, "I found it very helpful and wish more instructors used things like it." And finally, "I thought is was

a very useful tool in helping me out with this class, as well as understanding the terminology."

Joyce also had an evening class that did not receive the computer instructional aid. She did give those students the waiver/common final test the first day of class and, of course, gave the common final the last day of class. Also, since these students did not receive the computer instructional aid, she gave these students chapter review sheets that she had used before that she did not give to her day classes that received the computer instructional aid. Joyce gave me the pretest and posttest scores of her evening students, without names, for comparison with her day classes.

Table 4.27
6:00 Port Huron Evening Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Pretest	21	40	89	57.73	12.1963
Final Exam	21	67	99	89.14	9.2805

Although the program was not likely to make a significant difference in final exam scores, given all the variables that could be of influence, I wanted to check it as an additional "selling point." I compared the final exam scores of the two Port Huron classes (8:00 and 10:00) that received the computer instructional aid and the one Port Huron class (6:00 evening) that did not receive the computer instructional aid, and obtained the summary data in Table 4.28. For these comparison statistics, I only used the scores for the 35 students in 8:00 and 10:00 classes who had taken both the pretest and the posttest (common final exam).

Table 4.28
Comparing Port Huron Final Exam Scores

FINAL EXAM SCORES	Count	Low	High	Mean	Standard Deviation
Combined 8:00 & 10:00 classes that received aid	35	48	100	87.18	15.4868
6:00 class that did not receive aid	18	70	99	90.78	7.7198

Since the difference in the means was not in the desired direction, I decided not to conduct a t test. In examining the pretest scores and final exam scores for the 8:00 class in Table 4.9, the 10:00 class in Table 4.18 and the 6:00 class in Table 4.27, I noticed that the differences between the pretest means and final exam means seemed larger for the 8:00 and 10:00 classes (Table 4.29). The difference between the pretest and the final exam score would be a measurement of the student's improvement. Although the difference in the means was in the desired direction, it did not look very large. I decided to conduct a t test of the differences any ways.

Table 4.29
Comparing Port Huron Pretest to Final Exam Score Differences

FINAL EXAM MINUS PRETEST	Count	Low	High	Mean	Standard Deviation
Combined 8:00 & 10:00 classes that received aid	35	2	58	36.45	14.3838
6:00 class that did not receive aid	18	11	54	34.74	10.5513

My independent variable was availability of the computer instructional aid. My dependent variable was the difference between an individual student's pretest score and that student's final exam score. My hypothesis was that there was a treatment effect from the computer instructional aid. Therefore, my null hypothesis was that there was no treatment effect. In other words, there was no difference between the population means

from which these two sets of observations were taken. These are two-tail hypotheses. I used an alpha of .05. Unfortunately, subjects were not randomly assigned to the treatment, availability of the computer instructional aid. Therefore, the results could not support any interpretation that the independent variable caused the mean difference. Although I did not test the assumption of normality nor the homogeneity of variances, the t test is not sensitive to violations of this assumption (Shavelson, 1988). Since sample sizes were not equal, I used a pooled estimate of the population variance. The observed t was calculated as .445616. The critical t for a two-tail test with 51 degrees of freedom (35+18-2) was approximately 1.678. Since the observed t was less than or equal to the critical t, the null hypothesis could not be rejected. I could not show a significant difference between the treatment group's improvement from the pretest to the final and the control group's improvement from the pretest to the final.

Table 4.30
Post-Instructor Interview – Version 2

1. Did you like using the computerized questions and answers?
2. Why do you feel this was or was not a helpful tool for this course?
3. What problems did you encounter with the program?
4. What features of the program did you like?
5. What features of the program did you NOT like?
6. Any suggestions to make the program better?
7. Did you give students credit for using this program?
8. Would you like to use the program the next time you teach this course?
9. Do you think a similar tool would be helpful for other courses?

I had difficulty getting back to interview Joyce after the course ended. I did have a feedback form from her that I obtained during the pilot phase of this project but I wanted her reactions again at the end of fall 1998. Plus, there was the fact that I had changed the post-instructor interview questions. On January 28, 1999, Joyce emailed me her responses.

She said yes, she did like using the computer instructional aid because "it gave the students more exposure to the computer." She felt it was a helpful tool because "it gave the students actual test questions and gave them immediate response if they had the correct answer." She repeated the facts that "one student had problems saving to the disk because she used a different drive" and "there also was a problem of not always advancing to the next chapter when the student finished one chapter." The features she specifically liked were "the immediate response on the answer and the repeating of the questions they got wrong." She could not think of anything she did not like nor any suggestions to improve the program. She verified that she did give credit to her 8:00 class but not her 10:00 class. She said she "definitely" would like to use it again and that she thought it might be helpful in other classes. "My students think so, they have asked for it in other classes."

At the end of spring quarter, 1999, Joyce's dean decided to drop INF111, Information Systems Theory, on their campus. The dean had serious concerns regarding the importance of the subject matter covered in this course. She also felt they were receiving a large number of complaints from students who also questioned its importance. It was replaced with INF121, Windows, which did include some terminology.

Implementation Case #2: Judy on the Owosso campus

I interviewed Judy on Wednesday, September 30, 1998. This was the middle of week in the fall quarter. I had contacted her earlier and knew she was receptive to the idea. Actually, she started asking me questions before I could get to my planned

interview questions (Table 4.8 or Appendix L). She wanted to know if I expected her to require her students to use the computer instructional aid and turn in the printouts as an assignment. She explained that since the quarter had already started she had already given her students a list of assignments and tests and it would be difficult to change that at this time. I told her that it wasn't important. I told her that Joyce in Port Huron was already doing one class where it was required and one where it was not so I could get comparative data on that variable (see Implementation Case #1 above).

I then described the printout the students could receive at the end of each session if they requested it (Appendix N). She also wanted to know if the students had to run the entire test bank at once or could do it a chapter at a time. I explained that the main menu allowed the students to choose a chapter. "Then they can go back and do the chapter over again, or do another chapter or whatever they want."

In answer to my first interview question Judy said she did believe this was an important class but "not necessarily the way it's set up right now." She felt this was especially important for "students who've been out of school for 20 or 25 years, who never had any contact with" computers. She then went into detail explaining exactly what it was students needed out of a introductory course about a computer:

> Something that lets them know that they can, number one, learn it; and number two, they can be successful on it. I think that's necessary before they get hands-on but I'm not real convinced that our class, the way it's set up, is the best way to teach it. But I do think it's an important class from that aspect, that we still have students that are afraid of computers and we need to dispel those kind of fears and I think we can attempt to do that through a beginning class where they're not on a computer. So, I do think INF111 is an important class.

In answer to question 2 regarding the most important part of the class, she discussed the possibility of using the introduction to computers course as an assessment

tool. "We need to have some kind of an idea where the student is before we put them in an Excel class or a Word class but not necessarily so much Word as Excel where math skills are important." She then described the problem of trying to teach students how to use the electronic spreadsheet program Excel to do calculations when the students have weak math skills. She concluded by saying "I think we need to be able to assess some kind of basic skills."

In response to question 3, she stated clearly that she didn't like the common final. "I think it takes the responsibility of what you teach away from the instructor. I fit in and I try to teach the material so that they can pass the final exam. But I also tell them you'll probably learn something that's not in the final exam, in the process." I then reminded her that in an earlier discussion she had mentioned something about the difference between computer literacy and information literacy and asked her to explain what she meant. She said, "We can learn how to do one thing on a computer and learn to do it well." That's what she meant by computer literacy, the ability to do something on a computer. But just because we can do one thing on a computer only means we are literate in that area. She then went on to describe her views on information literacy.

> I think it's more important, to be able to use a computer to find information for whatever field it is you want. You can't be expected to remember everything that is thrown at you in four years for a degree or two years for a degree. But we should, at least, have been taught enough to know where to go to get the information. That's why I require my classes to go to the library and get an article off of Proquest or Infotrac or I don't care if it comes off the Internet if it indeed came off of a computer. So that they have some kind of an idea where they can go to get whatever information it is they need, on whatever subject. And I think that should be our main goal for their whole two or four years that they are here. Is that they know where to go to get the information, how to make sense out of the information.

When I asked her about any other aids or tools she gave her students to help them pass the final exam (question 4), Judy described a "scrambled final exam" that she uses. She mixed up the questions on the final and gave it to her students as a practice final exam. In fact, "if they pass that with a 94%, I don't make them take the final." She also made it clear that she does not use the Scantron machine to grade that practice test but hand grades it "because then I can see what questions they were having trouble with and know which one it is." On the advantages of using the computer instructional aid (question 5), Judy said she thought it would be helpful to students "because it gives them a different point of view. It gives them a different person asking the questions."

We then discussed the difficulties in using a computer room (question 6) and my using class time to distribute and demonstrate the program (question 7). Although Judy's classes did not meet in a computer room, there was a computer room available for her two day sections. Unfortunately, her 10:00 Monday and Wednesday section had 29 students in it and the computer room only had 24 computers. We knew that some students would have to double-up on computers. Her 4:00 Monday and Wednesday section only had 25 students so it wouldn't be as bad. We then made arrangements for me her classes to meet in the computer lab on Wednesday, October 7, 1998 so that I could introduce them to the computer instructional aid.

At question 8 we reviewed the fact that Judy had already given them students a list of all assignments for the quarter so she could not easily require students to use the computer instructional aid at this point. She did point out that "maybe we'll get a better result by making it available and not requiring it. At least we will see who's interested in learning more."

Judy was "real glad it doesn't just have the questions from the final exam" (question 9). However, at this point she just accepted the extra questions I had added and did not see a need to edit them. She also did not see any problems with letting students have a printout of the questions (question 10). She pointed out that she basically does that now when she gives them her practice final exam. "The only thing I require is that before they can take the final, they must turn that pre-test back in." She thought that giving them a printout of the questions would

> ...encourage them to go back through that book and look through these questions until in time they know what the answer is. So, it they're concerned about them having the questions, well, I don't know what the big secrecy is. But when you're doing that, how many students aren't going to recognize it any way if it's the same test they take to get out of that class. They're going to say, I read these before. Hopefully, maybe this time they know the answers to them. That's my philosophy.

In response to question 11 about printouts, Judy said she would primarily be interested in "what their final attempt was." Otherwise there was nothing in particular she would want the program to do (question 12) and no concerns or problems at this time (question 13).

Twenty-nine students in Judy's 10:00 class in Owosso received the program and signed the consent forms. Joyce's classes in Port Huron and Judy's classes in Owosso all had their final exams on the same day, Wednesday, October 21, 1998. I planned on distributing and collecting the survey forms to Joyce's 8:00 class. Then I was going to distribute the forms to her 10:00 but make arrangements for someone else to collect them and leave for Owosso. I wanted someone other than the instructor to collect the forms so the students would know that their responses would not affect their grades. Judy's 10:00 class in Owosso was not taking their final exam until about 10:45. I mistakenly thought I

could get to Owosso before the students completed their final and distribute the student survey form. Unfortunately I was overly optimistic and many of Judy's students had left by the time I got to Owosso. I was only able to collect five survey forms from her students. Judy did have the students turn in their floppy disks so I was able to collect data from them and obtain their final exam scores (Table 4.31).

Table 4.31
10:00 Owosso Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Final Exam	29	64	100	90.21	8.1688

The responses to the first question on the survey form are show in Table 4.32. One student added the comment, "It may help if we could be in the computer lab some part of the course."

Table 4.32
10:00 Owosso Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	5				

The second question related to ease of use (Table 4.33). One student described the problem of "remembering where everything was placed such as RAM and ROM, and auxiliary storage, secondary storage. Those types of things, but I got the hang of it." I suspect this student may have misunderstood and was commenting on the course rather than the software. Another students comments more directly related to the software. She wrote, "Not being that involved in computers I did have some difficulty. Some simple errors. Then when taking it home and trying to get to the program it was not so easy."

Table 4.33
10:00 Owosso Class - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	2	3			

Question 3 asked students why they used the program (Table 4.34). The same student who seemed to be commenting on the course in question two, gave a comment here that also seemed to relate to the course: "It was part of my major, plus it helps to learn a little of the same things over again." This response was not categorized in Table 4.34.

Table 4.34
10:00 Owosso Class - Student Survey Question 3

STUDENT SURVEY QUESTION3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? What influenced you?	1	1	1	1	

All five students rated instruction in using the program as "Very Good" (Table 4.35). Since this class was able to receive hands-on experience using the program in class, this was not surprising. One student did add the request to "Please slow down in the very beginning, as not all students know how to use and turn on the computer. Please slow down to get everyone to first base. Thanks."

Table 4.35
10:00 Owosso Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	5				

Two of the students did not answer any questions of the back of the form so only three students responded to questions 5 to 8. The students may not have noticed the

questions on the back. Of the three students who answered question five, several different places were marked (Table 4.36). The student who marked "Other" described this place as "lab- re. class." This might be referring to the class time we spent working with the program.

Table 4.36
10:00 Owosso Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	2		1	2		1

In answering question 6 regarding chapters used (Table 4.37) one student marked only chapter 2. This was very unusual. Even those students who did not use all chapters almost always marked all chapters up to the last one he or she used. For example, typically a student would have marked both chapters one and two.

Table 4.37
10:00 Owosso Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
6. The software covered 7 chapters. Please check all you used:		1					2

The same student who seemed to be evaluating the course rather than the program in questions 2 and 3 above all provided a strange comment to question 7. The complete question was "Think back over all of your use of the software. How many hours total did you use the software?" The comment this student made was "Ever since the ninth grade in my first computer class we always used this in the high school." Perhaps this student was referring to how long she had used similar software or any software.

Table 4.38
10:00 Owosso Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?			2								

Table 4.39
10:00 Owosso Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?	3	2	2	2	2	0

Only one student responded to question 8 asking for suggestions. That student's comment was "I just wish to say that it's easy to understand and to read. Please in the initial stage explain things slower. Thanks." This was the same student who requested I "slow down in the very beginning" in answer to question four above.

Twenty-five students in Judy's 4:00 Owosso class received the computer instructional aid and signed a consent form. Only 21 students completed a student survey form including two students who specifically stated that they did not use the program other than during my introduction in class.

Table 4.40
4:00 Owosso Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Final Exam	24	65	100	91.67	9.2762

In answer to question 1 regarding usefulness of the software (Table 4.41), one student did not mark a choice but stated, "Didn't use it very much." Her other answers indicated she did not use the program outside of class. I choose to count this as not applicable (NA). Two other students marked NA and later in the form also indicated that

they did not use the computer instructional aid outside of class. The comment area under this question was titled "Suggestions for making the program more helpful." One student said, "When you don't get a question [correct] the first time, change it a little to make it different when asked again." Another student suggested, "Not making boring."

Table 4.41
4:00 Owosso Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	8	8	2		3

Many students indicated they found the software easy to use (question 2, Table 4.42). Three even wrote comments such as "none" after the request for problem descriptions. Problems mentioned included "It wouldn't let me look past chapter 2" and "I noticed a lot of bugs." One student described her problem in detail: "Every time I wanted to switch chapters I had to type in GWBASIC INF111." GWBASIC INF111 refers to the DOS command to start and load the program. Another student may have been referring to the same problem when she stated "Forgetting the word INF111."

Table 4.42
4:00 Owosso Class - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	13	7			1

There was a wide range of responses to question 3 asking students why they used the program. There were six responses that I could not categorize. These included "Because I need all the help I can get when it comes to studying," "Class," "Study guide," and "Not very much." Two students who marked question one NA, apparently because they did not use the program very much, added the comments "Went during class" and "I used it once and I was curious what it was like."

Table 4.43
4:00 Owosso Class - Student Survey Question 3

STUDENT SURVEY QUESTION 3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? What influenced you?		8	5	1	3

Question 4 related to instructions using the computer instructional aid. One student commented, "Once in the program, everything is pretty self explanatory." Another student who marked "Very Good" mentioned, "a sheet of instructions were received with the disk." There were about three or four students in the study who were absent the day of the demonstration and received the floppy disk containing the program and the sheet of instructions for running the program from their instructor. These students were not specifically identified but this may have been one of those students.

Table 4.44
4:00 Owosso Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	13	4	3		1

Two of the students left the entire back of the survey form blank. One of the students who reported not using the program responded with NA to all of the questions on the back. As a result these students were not included in tabulations for questions 5, 6 and 7. Only one student marked two answers to question number 5 regarding where the program was used (Table 4.45).

Table 4.45
4:00 Owosso Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	4	1	1	13		

In regards to how many chapters were used (question 6), many students indicated they had only used chapter one (Table 4.46). Two students specifically stated that they only used the program for the hands-on demonstration during class time.

Table 4.46
4:00 Owosso Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
The software covered 7 chapters. Please check all you used:	7	3					8

Using the same processing procedures as for other class sections I tabulated the responses for question 7 (Table 4.47). Four students put NA or left the answer blank. One student responded "4 times or more, I don't know." These last five students were not included. Commenting on the number of hours, one student said, "It would have been very useful if I used it more." Another student who indicated one hour of usage admitted, "The only time I used it was during my class."

Table 4.47
4:00 Owosso Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?	2	8	4	1							1

Table 4.48
4:00 Owosso Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?	5	16	.675	10	1.98	2.2305

Question 8 was the final open-ended question encouraging suggestions and criticisms. Responses included "I think the software is fine the way it is," "This program

is boring," "Some of the questions were too long," and "This was a helpful thing I liked it." One student stated, "If you give it to the class from the beginning." This was a reference to the fact that the Owosso students only received the program two weeks before the final exam. Several students repeated problems they had mentioned earlier on the form: "This program wouldn't allow me to look past chapter 2"; "Make it easier to go from chapter to chapter, rather than quit and go back into the program"; "Just try and get the bugs out. It would be very helpful then." One long comment explained the student's situation and why she had only used one chapter and only spent an hour and forty-five minutes using the program:

> I didn't find any fault with the program. However due to my transportation situation for that week I couldn't get to the lab and I don't have a computer system at home.

My interview with Judy at the end of the course was rather short. I only managed to catch her for a few minutes between classes. She did feel that the computer instructional aid would be "beneficial if students were in a computer lab. But when there are 40 to 50 students in a class, you cannot easily take them to a computer lab that only contains 24 computers." She stated that she liked the fact that the program gave the students the "opportunity to try again" when they missed a question. It "doesn't kick them out. "She said that she would like the program to be "capable of printing out questions so students can find the answer in the textbook and write it down." She also mentioned that some of the questions on the exam are different from what is in the new textbook we had just switched to.

Implementation Case #3: Tina on the Muskegon campus

I called the academic dean on the Muskegon campus that is responsible for the INF courses. He suggested I talk to Tina who teaches most of those courses and had taught them for a couple of years. On September 17, 1998, before the quarter began, I emailed Tina and explained to her what I wanted to do and asked if you would be willing to participate in my study. As it turned out, she was teaching a section of INF111 in the second half of the quarter on her campus. Most sections of INF111 are offered in the first half of each quarter. Only on the larger campuses do administrators sometimes schedule one or two sections in the second half. Since I was already working with Joyce in Port Huron and Judy in Owosso during the first five weeks of the quarter I decided it would be easier to work with Tina during the second five weeks. On October 14 I emailed her and asked if we could arrange to meet. On October 17th she emailed me back and said she would be in Flint on Sunday, October 25th if that would be more convenient. We met in the afternoon in the library on the Baker College Flint campus.

Before I started my interview questions (Table 4.8 or Appendix L) I checked to see if she had taught this class before. She explained that she started teaching it about seven years ago when it was in a different format. She said she liked teaching it, in fact, she preferred it to the hands-on computer classes. I gave her a copy of the interview questions and we started working through them. She said she did think it was an important course (question 1) but "something about it doesn't seem quite right because we get the students in the class and some of is important but I don't think it's important for everyone." She went on to discuss different viewpoints: "Do I possibly need to know that as a user, the term word size, and exactly what it means? I don't. So I think some of that

maybe gets too technical." Although we were still on question 1 her comments actually started relating to the common final (question 3). "Some of the topics for discussion are good but they have no relevance to what we test on. So if I spend too much time on that and no say you have to know this term and this term and this term" students do not learn what they need to know to pass the final.

In answering question 2 relating to what is important she described what areas she thought were important but were not covered by the course. For example, she felt the course should be more "application oriented in terms of how the computer has affected our everyday life maybe. And I try and have a lot of discussion like that but then again, we spend a whole day talking about stuff like that and then we don't memorize all of the terms." She did say that "some of the terms, I think, are good to know because if a person is going to shop for a computer" they need to know them.

When asked if she was satisfied with the common final exam (question 3) she discussed her complaints by comparing it with an earlier version. She was concerned that the common final "got more technical in some areas... like what is the formatting step in word processing?" "When I read some of those software specific questions I couldn't believe it because they're not using the software." "They've got to be kidding to be that specific." She suggested that the writers of the common final should give the test to "10 or 12 instructors who weren't part of the process to take it and see what they think." In addition to the actual terms that were on the final, she was also concerned with the writing. "Some of the questions it's the way it's worded that can be quite difficult depending on the person. I think multiple choice have to be a little more straightforward."

Question 4 asked about study aids and tools. Tina said, "I give quizzes every week over two chapters." She went on to explain, "On those quizzes I give 15 multiple choice questions. They're all very similar to the kinds of questions on the final." She then described in detail several games and small group activities she uses to review and practice for the final.

When asked if she thought the computer instructional aid would help students (question 5) Tina aid "Yeah. I think so. It's just [that it would] depend on if the students had access to the computer to do it.... That's something they'd have to do on their own and not all students are going to do that."

Tina explained that it is "fairly difficult at our campus to use a computer during class time" so we would not have access to a computer room (question 6). But when I asked if I could use some class time (question 7) she said, "Oh yeah, sure. That's an easy one." I explained I would make arrangements to use a portable computer with and LCD panel for an overhead projector so I could show the computer instructional aid to her students. We then discussed the possibility of incorporating the aid into her grading system (question 8) and she was quite willing to add it to the list of things she lets students do for extra credit.

My next group of interview questions related to features of the computer instructional aid. The exact questions on the test bank were not important to Tina (question 9). "Because anything that they'd do would be helpful: the ones from the final or ones that are similar type questions." She did not think students should be allowed to print out the questions in the test bank (question 10). She also thought that we should not disclose the fact that the test bank contained the actual questions on the common final.

"Obviously tell them these are the kinds of questions you are going to have on the final, some might be very similar." She said the existing printouts from the computer instructional aid would be sufficient. She thought it was important that students be able to identify areas they were having problems with. "If they already see the chapters and what sections they had trouble with, that is enough to go out on their own" to study.

She did make a suggestion (question 12) regarding the possibility that the program could "track how many students were missing certain questions." She added that since students were being allowed to run the program anywhere, including on home computers, she knew there was no way to easily collect and tabulate that information. In regards to problems and concerns (question 13) we returned to her dissatisfactions with the common final and her difficulty in becoming involved as a part-time instructor. She said, "There's hardly any full-time people. I mean, most people are part-time and if they are, then they have other responsibilities, other projects, other jobs, other whatever, that this is not their focus."

I visited her class on Tuesday, November 3, 1998. That was week seven of the quarter or week two of the class since it was held the second five weeks of the quarter. Thirteen students were in class that day and all of them signed the consent form and received a floppy disk containing the computer instructional aid. This version of the computer instructional aid included a fix to the "bug" found by the Port Huron and Owosso students that sometimes prevented the student from moving from one chapter to another. The two errors in the test bank had also been fixed. Five students were not there but signed a consent form and received the program from the instructor at a later class session. Tina's class did not meet in a computer room and one was not available during

her class period. However, I was able to use a computer with and LCD projection panel and overhead projector so all of the students could see the monitor. I demonstrated running the computer instructional aid using the printed instructions that each student was given.

Of the total 18 students, six students either dropped or just did not show up for the final. As a result, I had final exam scores for only 12 students (Table 4.49) and of those, one student did not complete a survey form. Of the 11 survey forms, two students did not answer any of the questions but simply wrote a short note stating they had not had enough time to use the program.

Table 4.49
10:00 Muskegon Class, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Final Exam	12	51	99	79.75	16.1083

Only one student added a comment below question one on the student survey (Table 4.50). That student indicated the software was "Very Useful" and added the comment: "If you have the time to use it."

Table 4.50
10:00 Muskegon Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	7	2			

In response to question two (Table 4.51) regarding ease of use, one student who marked it as "Easy" wrote that the program "locked up after chapter 5 (asking to do chapter 6)." The student who found it "Very Difficult" to use explained that "on my home computer, would not run. Said something like Not in Shell?" The computer instructional

aid was designed to run in DOS. In the Windows environment this is supposed to operate in a special window or "shell."

Table 4.51
10:00 Muskegon Class - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	5	3		1	

Although Tina did not require her students to use the computer instructional aid, she did give them extra credit if they used it. Therefore if a student mentioned extra credit in response to question 3 (Table 4.52), I categorized this as the instructor influenced the student. There were two interesting comments that I was unable to categorize. One student wrote, "I used it to see how good I was doing on the chapters." Another simply wrote, "For the exam."

Table 4.52
10:00 Muskegon Class - Student Survey Question 3

STUDENT SURVEY QUESTION 3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? what influenced you?	3	5	1	2	

Compared with earlier classes, responses to question 4 (Table 4.53) were somewhat low. This was to be expected since I was only able to demonstrate the computer instructional aid, not allow them to actually work with it, during class time.

Table 4.53
10:00 Muskegon Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	6		3		

Three students did not answer any questions on the back, leaving only six students responding to questions 5 through 8.

Table 4.54
10:00 Muskegon Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	4	1		1		

Table 4.54
10:00 Muskegon Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
The software covered 7 chapters. Please check all you used:	1		1				4

I used the same rules as before in summarizing the answers to question 7. For tabulation purposes (Table 4.56), I dropped fractions and used the lowest value in a range. For statistical summary purposes (Table 4.57), I converted fractions to decimals and used the middle value of a range. For both tables, I ignored qualifying statements such as "about." Port Huron and Owosso students could have indicated that they did not use it (0 hours) outside of class but still make useful comments on the survey form because they did use it for a short time in class. Since Muskegon (and Flint, see below) did not use it in class, students who did not use the program at all did not answer any of the questions on the survey form. So, although two students wrote notes on their forms stating they did not use the computer instructional aid, neither of them filled out the form nor answered question 7 and are not included here. The student who used the program three hours added the comment that she "would have liked to use more but just didn't have the time." This was the same student who made a comment about time in regards to question 1 above.

Table 4.56
10:00 Muskegon Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?		1	2	1	1				1		

Table 4.57
10:00 Muskegon Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?		6	1	8	3.54	2.3896

Since I corrected the program after receiving the email message from Joyce on the Port Huron campus on October 12th before these students received the program, only one Muskegon student mentioned specific problems using the computer instructional aid (see student survey question two above). In response to question 8's request for suggestions and criticisms, one student wrote, "It was pretty good." Another student responded positively with "It thought it was fine the way it was." A third student wrote the comment, "I felt if the questions would have been scrambled each time within the chapter it would have been more challenging." Since the questions were supposed to be randomly presented I never did understand this comment unless the program was working properly for some reason.

On December 29th, 1998, Tina emailed me her responses to my post instructor questions (Table 4.30 or Appendix M). She wrote that yes, she did like the computerized questions and answers. She said she thought it was helpful because, "it seems the students don't really like the wording/style of the final questions - so practicing them on their own

is very helpful." She mentioned that one of the problems she encountered was that "a few students indicated that they could not get the results to print."

The feature she liked the most was "asking the questions until answered correctly, then repeating them." There was "nothing" she did not like. Her suggestions included "checking the printing options," "a random 'quiz' as well for students to choose to practice," and "possibly putting the program 'online' to make it easier to use from a web site." As I knew, she did give extra credit points for using the computer instructional aid. She wrote that yes, she would like to use the program next time she taught this course and yes, she thought a similar tool would be helpful for other courses.

Tina was a part-time instructor and unfortunately did not return in the fall of 1999.

Implementation Case #4: Cynthia on the Flint campus

On September 24, 1998 I sent an email message to an instructor I knew on the Flint campus. I had noticed, on the schedule, that she was one of the instructors teaching INF111 in the second half of the fall 1998 quarter. I already knew I would be working with Joyce in Port Huron and Judy in Owosso during the first half so I was hoping to find another instructor on a different campus that I could work with during the last half of the quarter. The Flint campus has the most students and had several sections of INF111 in the second half of the quarter. The Flint campus was also relatively close to me. As it happened, the instructor I knew was not teaching the class at a time when it would be convenient for me to visit her class. She did put me in touch with Cynthia who was teaching the class on Monday evening when I did not have any conflicts. There was also

the factor that all of the other classes I was working with were day classes and I thought it might be interesting to include an evening class.

It turned out that Cynthia was working during the day at another job and teaching part-time at night so it was difficult to meet with her. I finally settled for sending her my Pre-Instructor Interview questions by email (Table.4.8 or Appendix L) and she emailed her answers back to me on November 26. This was actually after I had already met with her students and distributed the program. She wrote that yes, she did think this was an important course "but not in its present format. The assigned textbook gears students for lab assignments which I think would be beneficial to assisting students with so much new technology." On the Flint campus, INF111 classes are always taught in a classroom, never in a computer room, because of the limited number of computer rooms. She went on to explain that she thought the most important part of the class was the "students' discovery of the world of technology about them and its expansiveness."

She thought that the common final seemed appropriate. To help her students pass the final exam she has used:

> Study guides/outlines for chapters one through five which required students to write in definitions of terms for reinforcement. Word search and crossword puzzles were used. Also, a competitive game of Jeopardy for fun and reinforcement of chapter terms.

She did think the computer instructional aid would help her students: "Very much so."

She did confirm the problems I suspected, in using a computer room during class time. She said it would be "extremely difficult" since "class size is normally 20-45 students" and the "learning center cannot accommodate" and computer rooms "(though in use) seat 23 max." However I could use class time to distribute and demonstrate the program and she could require students to turn in the printout indicating they used the

computer program. As it turned out, she did not give require students to use it or give them credit.

She did not have any questions she wanted in the program but she did think students should be able to print out the questions. In fact, in terms of results that students could print out at the end of the program, she specifically mentioned "questions and answers of items answered incorrectly." She did not have any suggestions for anything else the program should do.

Her only concern was:

> This is an entry level course that should not be too rigorous that students feel they won't be successful in their other applications courses. All students (particularly the novices) should walk away after five weeks feeling much more confident, knowledgeable, and empowered about their inevitable encounters with technology.

I visited Cynthia's class on Monday, November 16, 1998. Although this was Monday of the fourth week of the class (ninth week of the quarter) this was two weeks before the final exam since Thanksgiving week fell between weeks 4 and 5. I distributed the computer instructional aid and instructions to 25 students who signed consent forms. This version of the computer instructional aid and instructions included a minor change so that students did not need to enter responses in upper case as mentioned earlier in the section on creating the instruction sheet. Since we were not able to use a computer room, I used a portable computer with an LCD projection panel and overhead projector to demonstrate running the computer instructional aid. I encouraged students to read the instructions as I performed each step.

On Monday, November 30, I visited the class again. I handed out the survey form at the same time Cynthia handed out the final exam and then waited until they were done.

When students finished, they turned in the final exam to the instructor and gave me their completed survey forms. Several students did not turn in survey forms. I was not surprised since I always made it clear that using the computer instructional aid and filling out the survey form were completely voluntary. Twenty-four students took the final exam and 18 students turned in survey forms. Five of the students who turned in forms did not use the program.

Table 4.58
6:00 Flint, Test Results

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Final Exam	24	62	100	87.63	10.9735

The responses to the first question on the survey form are show in Table 4.59. The request for comments was titled "Suggestions for making the program more helpful." Of the students who marked "Very Useful" one student added the comment, "Good one?" Overall, this student rated the program fairly well so I was not sure whether the student was suggesting I make the program a "good one" or was stating that the program was a "good one." The presence of the question mark did not help my interpretation of the comment. Another student who marked "Very Useful" wrote "Have some questions that don't pop up the first time you go through it that way it will be more changing." A third student who marked question one as "Very Useful" suggested "windows accessible or on-line." Only one student who marked it as "Somewhat Useful" added a suggestion. He wrote "color background, use a mouse." The student who indicated the software was only "a little bit" useful wrote "able to do all chapters not just one." When this student turned in the form she told me she was only been able to do one chapter. She also wrote "only did one chapter, wouldn't let me do the others" after question number 2, below. A student

who did not use the program marked NA for both questions 1 and 2. He wrote "instructions for Windows 3.1" after question one and after question two he wrote "I tried to get it to work, I typed A: and then it displayed same black screen similar to yours. I had to push CNTRL + ALT + DEL."

Table 4.59
6:00 Flint Class - Student Survey Question 1

STUDENT SURVEY QUESTION 1	Very Useful	Somewhat Useful	Little Bit	Not At All	NA
How useful was the software?	8	4	1		

Comments from students who indicated the software was "Very Easy" (question 2, Table 4.60) included:

> One error- question on program that controls hardware, time, etc. Correct answer- operating system. Disk said integrated software.
> I had no problems.
> No problems using or printing.

A student who marked the software as "Somewhat Difficult" to use wrote: "Would not always give correct answer, then when I would put in the answer it choose before, it still came up wrong."

Table 4.60
6:00 Flint - Student Survey Question 2

STUDENT SURVEY QUESTION 2	Very Easy	Easy	Somewhat difficult	Very Difficult	NA
How easy was the software to use?	6	4	3		

The responses to open-ended question number 3 (Table 4.61) seemed extremely hard to categorize for this class. The one student I categorized as being influenced by requirements or instructor, actually wrote "you, the school." The students that I categorized as influenced by expectations wrote:

> To help prepare for the test.
> To raise my grade.

The students who simply indicated they used it to study wrote:

> For final - better grades and time efficient.
> Final exam preparation.

I was completely unable to categorize the responses of eight students. Their comments were:

> I was influenced because the disk had our final on it and I needed the help.
> I wanted an A.
> To see if it would improve test scores.
> Study aid.
> Different form of study - break from a book!
> I learn more by repetition.
> I was interested in using a computer for first time.
> I thought it would be more helpful than the book because I could answer questions instead of sorting through page after page.

The student who was not able to get the program to work and marked NA for most questions did write after this question:

> I wanted to get a good grade.

Table 4.61
6:00 Flint Class - Student Survey Question 3

STUDENT SURVEY QUESTION 3	Required (Instructor)	Expected it to help	Found it helped	To study	No answer
Why did you use the program? What influenced you?	1	2		2	

Question 4 related to instruction (Table 4.62). Several students who marked it "Very Good" used this comment area to make general comments about the program:

> Over all I think that this is an excellent way to study. Repetition stores info in our own main memory.
> This helped me so much on the final!
> Thanks.

One student did comment directly on the instruction in using the program. She wrote:

"When explaining how to use it you used the most simple terms possible." The student

who was unable to get the program to work put an additional note here: "I tried to use it by going into windows and going to main and then MS-DOS."

Table 4.62
6:00 Flint Class - Student Survey Question 4

STUDENT SURVEY QUESTION 4	Very Good	Good	Adequate	Inadequate	NA
Did you receive adequate instruction in how to use the program?	10	2	1		

Of the 13 students who used the program and filled out the survey form, one did not answer any questions on the back. Each of the remaining 12 students only marked one answer to question number 5 (Table 4.63). The table does not include the one student who was unable to get the program to work but filled out the survey form and marked "Home" as the location where he used the program.

Table 4.63
6:00 Flint Class - Student Survey Question 5

STUDENT SURVEY QUESTION 5	Home	Friend or Relative	Work	Baker Lab	LSS	Other
Where did you use the program?	10		1	1		

One student only indicated that she had only done one chapter, in answering question number 6 (Table 4.64). This is the same student who commented after question two that the program would not let her do any other chapters.

Table 4.64
6:00 Flint Class - Student Survey Question 6

STUDENT SURVEY QUESTION 6	1	2	3	4	5	6	7
The software covered 7 chapters. Please check all you used:	1				2		9

I used the same processes in determining the following values for question 7 relating to hours (Tables 4.65 and 4.66) as I used for the other classes. Unfortunately, 2 of the 12 students who used the program and answered questions on the back did not answer this question. One wrote, "Don't remember." The other just wrote, "Would of loved to use it the whole five weeks to help with tests." Some interesting comments included the student who indicated she had used it about ten hours wrote, "The more I used - the better I got!" Another student just wrote "I like it." The student who was unable to get the program to work wrote, "I tried to get it to work. I spent 20 minutes and then I gave up!"

Table 4.65
6:00 Flint Class - Student Survey Question 7

STUDENT SURVEY QUESTION 7	0	1	2	3	4	5	6	7	8	9	10
How many hours?		3	2	1	1	1				1	1

Table 4.66
6:00 Flint Class - Student Survey Question 7 - Statistical Summary

STUDENT SURVEY QUESTION 7	No Answer	Count	Low	High	Mean	Standard Deviation
How many hours total did you use the software?		10	1	10	3.95	3.3620

General comments and suggestions in response to question eight included:

I think it was put together very well!
Some questions were too easy. A higher thinking level would be beneficial.
Colored background, use center of screen, move mouse pointer so user can get farther from screen.
User interface would be nice but not necessary.
I love it! I only wish I had one for every class.
A lot of the test questions were the disk questions right down to A, B, C, D answers. I think that should be changed because I could of just remembered the letters instead of the actual answer.

When I distributed and collected student survey forms, I also gave Cynthia a copy of my post interview questions (Table 4.30 or Appendix M). Actually, she ended up writing out the answers to several of them. When asked if she liked the computer instructional aid, she wrote "Very much!" In response to question two regarding why she thought it was helpful, she wrote "Repetition was the key to retention of terminology." She did not encounter any problems with the program. The features she liked were "repeat question, print, summary." She did not give the students credit for using it. When I asked if she would like to use the program next time, she responded "Yes! May we use it?" Of course, I gave her permission right away. Finally, she did think this could be helpful in other courses, "particularly where new terms are incorporated in multiple choice tests."

In April of 1999, between winter and spring quarters, Cynthia emailed me that she had offered it to students in the winter quarter of 1999. She said about six students "took advantage of it and did well." She went on to say:

> I just called Debbie ... Tuesday and left a message that I wanted to refer your program to her. This is her first term teaching INF111. I'll be looking forward to your web site next week.

The last statement referred to the fact that I had emailed her that I was setting it up so the last version of the program could be downloaded from my personal web site.

Post-Implementation

Evaluation

The goal of this project is to document and compare an insider's developing and diffusion of an instructional aid to standard approaches using an outsider. One goal of the diffusion process is to get clients to accept the product. One means to acceptance is to have clients see the product as useful. In this case I wanted to determine if instructors and students found the computer instructional aid useful. This also relates to requirement number 5 identified at the beginning of the actual development of the computer instructional aid: The program needed to appear to be something the student would find useful.

Another means of acceptance is to see if the product could be easily integrated into the existing course as taught by different instructors. As an extra possible selling point, I also checked to see if use of the computer instructional aid had any statistically significant impact on student grades on the common final exam. But before examining these issues, I first evaluated the computer instructional aid in terms of the original requirements identified for it.

Program Requirements

During the winter and spring of 1999 I evaluated the computer instructional aid. I started by reviewing the requirements I had identified at the beginning of the actual development of the program. (1) I had included the actual questions from the common final exam in the test bank used by the computer instructional aid, in multiple choice format as they appeared on the exam. (2) The program did provide immediate feedback

to the student's entries by stating correct or incorrect. (4) The program was completely student controlled. In response to any question, the student could enter a "Q" to indicate quit and the program would return to the main menu where the student could exit the program, restart the chapter, or start another chapter. The student was also returned to this same menu at the end of each chapter where he or she had the same options. From that same menu, the student could also control if and when output would be printed.

Item (3) in the original requirements was "the program needed to be easy to use." Since the students were the final users it was important that the students find the program easy to use. This was a substantial goal since this was a beginning computer course and most students had little, if any, prior computer experience. Question 2 on the student survey form dealt directly with this issue (Tables 4.67 and 4.68).

Table 4.67
Student Survey Question 2 by Campus

How easy was the software to use?	Very Easy		Easy		Somewhat difficult		Very Difficult		Total
Port Huron - 8:00	10	50%	8	40%	2	10%			20
Port Huron - 10:00	13	81%	3	19%					16
Owosso - 10:00	2	40%	3	60%					5
Owosso - 4:00	13	65%	7	35%					20
Muskegon - 10:00	5	56%	3	33%			1	11%	9
Flint - 6:00	6	46%	4	31%	3	23%			13
All sections	49	59%	28	34%	5	6%	1	1%	83

As shown in Table 4.67, 93% of all students responding to this question on the student survey form reported the program was "Very Easy" or "Easy" to use. Some factors that could have affected these scores were (1) the design of the computer instructional aid itself, (2) the instruction sheet for students, and (3) the hands-on session or instructor demonstration in how to use the program. Table 4.68 compares the responses of those students introduced to the program in a hands-on session in a

computer lab (Port Huron and Owosso students) with those students introduced to the program by an instructor demonstration (Muskegon and Flint students). As might be expected, a larger percentage of students introduced in a lab responded "Very Easy" or "Easy" than those students introduced with an instructor demonstration. Students introduced in a lab would find it easier to identify and correct misconceptions in the proper use of the computer instructional aid than students only receiving a demonstration.

Table 4.68
Student Survey Question 2 by Introduction

How easy was the software to use?	Very Easy		Easy		Somewhat difficult		Very Difficult		Total
Lab	38	62%	21	34%	2	3%			61
Demo	11	50%	7	32%	3	14%	1	5%	22

Question 4on the student survey form also addressed, indirectly, original requirement number three: "the program needed to be easy to use." Although only one of the three factors I identified above as influencing ease of use, the hands-on or instructor demonstration, was the most obvious form of "instruction in how to use the program." Table 4.69 shows the summary of responses to question 4.

Table 4.69
Student Survey Question 4 by Campus

Did you receive adequate instruction in how to use the program?	Very Good		Good		Adequate		Inadequate		Total
Port Huron - 8:00	19	90%	2	10%					21
Port Huron - 10:00	13	81%	1	6%	1	6%	1	6%	16
Owosso - 10:00	5	100%							5
Owosso - 4:00	13	65%	4	20%	3	15%			20
Muskegon - 10:00	6	67%			3	33%			9
Flint - 6:00	10	77%	2	15%	1	8%			13
All sections	66	79%	9	11%	8	10%	1	1%	84

As shown in Table 4.69, 90% of all students who responded to this question on the student survey form indicated that the instruction was "Very Good" or "Good."

Conversely, only 1%, actually only one student, found the instruction "Inadequate." Although some students commented that the instruction could have been slower, this seems to show a very positive response to the instruction in the use of the computer instructional aid.

Comparing the responses for received instruction in a hands-on session in a computer room with those students who received only a demonstration, those who received instruction in a lab, did rate the instruction higher (Table 4.70). 92% of those receiving instruction in a lab rated the instruction "Very Good" or "Good," while only 82% of those receiving instruction using a demonstration rated the instruction this high. This would be expected since the hands-on instruction would more closely match the students later experiences working with the computer instructional aid than a demonstration.

Table 4.70
Student Survey Question 4 by Introduction

4. Did you receive adequate instruction in how to use the program?	Very Good		Good		Adequate		Inadequate		Total
Lab	50	81%	7	11%	4	6%	1	2%	62
Demo	16	73%	2	9%	4	18%			22

Usefulness

One of my major goals of the student survey forms and instructor interviews, was to find out if instructors and students, especially, would find the computer instructional aid useful (requirement number five). In both cases, perceptions alone are very important. The student or instructor starts with his or her anticipation or expectations of usefulness. As the student starts to use the computer instructional aid, those expectations are either supported or contradicted.

Usefulness to Students – Perceived

As mentioned above, a student's anticipated or expected usefulness can be very important. The student's instructor, the outside presenter (myself), and the description of the computer instructional aid influence these early perceptions. If there is little expected usefulness, the student may not use the aid unless there is an outside influence such as a course requirement or extra credit. Those students who participated in a hands-on session in a computer room were placed in a situation where they were expected to use the program. It would have been very difficult for a student in that situation to refuse to use the program during that session.

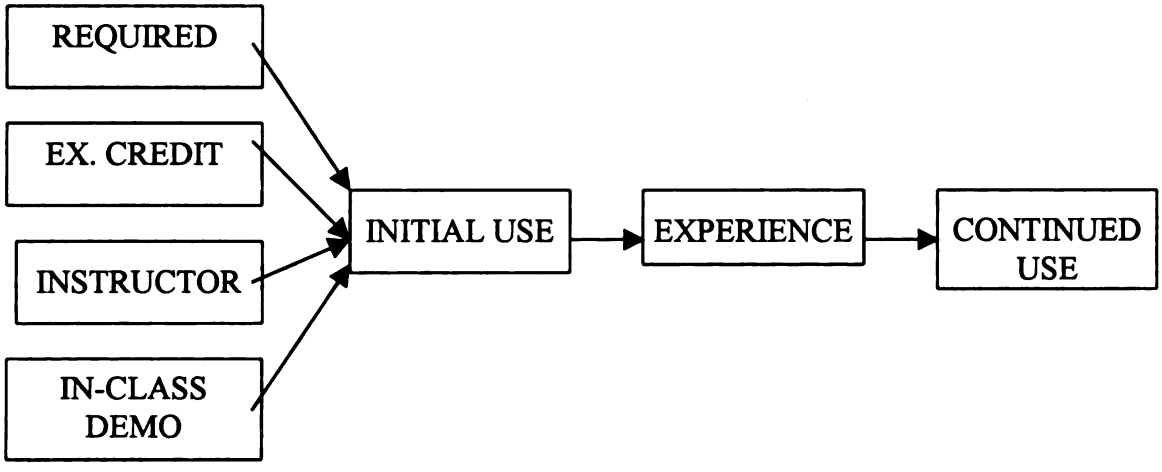

Figure 4.4. Factors Affecting Use

Once the student has used the program, his or her perceptions are modified by that use as mentioned above (Figure 4.4). The major influence on this modification of student perceptions of usefulness is the computer instructional aid itself. If a student's use of the aid does match the student's expectations of the upcoming final exam, the student will continue to believe the aid is useful and continue to use it. Of course, the final test of usefulness is after the final exam. If, at that time, the student still feels the aid was useful,

then the computer instructional aid did provide the student with an experience that the student felt benefited his or her performance on the final exam. This was the reason students were encouraged to complete the student survey form after they had finished the final exam.

Question 3 on the student survey form was an open-ended question that tried to identify what influenced the student to use the computer instructional aid (Table 4.71). This question primarily addresses initial use. The largest group used the program because they found it helped. The Port Huron 8:00 class was required by the instructor to use the program. The Muskegon 10:00 received extra credit for using the program. As I would expect, a larger percentage of these classes mentioned requirements, credit or their instructor, than other classes (Table 4.72). However, a much larger percentage of the students for whom it was not required, mentioned they expected or anticipated that the computer instructional aid would help.

Table 4.71
Student Survey Question 3 by Campus

Why did you use the program? What influenced you?	Required (Instructor)		Expected it to help		Found it helped		To study		Total
Port Huron - 8:00	5	23%			12	55%	5	23%	22
Port Huron - 10:00	2	12%	3	18%	7	41%	5	29%	17
Owosso - 10:00	1	25%	1	25%	1	25%	1	25%	4
Owosso - 4:00			8	57%	5	36%	1	7%	14
Muskegon - 10:00	3	27%	5	45%	1	9%	2	18%	11
Flint - 6:00	1	20%	2	40%			2	40%	5
All sections	12	16%	19	26%	26	36%	16	22%	73

Table 4.72
Student Survey Question 3 by Credit

Why did you use the program? What influenced you	Required (Instructor)		Expected it to help		Found it helped		To study		Total
Not Required	4	10%	14	36%	13	33%	9	23%	40
Required/Credit	8	24%	5	15%	13	39%	7	21%	33

I used questions 6 and 7 of the student survey form to identify continued perceptions of usefulness. If, after initial use, students continue to use the computer instructional aid, this would indicate their perceptions of usefulness were supported by their experiences using the aid. Of those responding to question 6, about three-quarters (74%) indicated they used all seven chapters (Table 4.73). Columns 4 and 6 were omitted from Table 4.73 because no student reported completing only four or six chapters. Interestingly, almost the same percentage of the 8:00 and 10:00 Port Huron classes reported using all seven chapters in spite of the fact that the 8:00 was required to use computer instructional aid as an assignment and the 10:00 was not required. Statistical summary data for question 6 is shown in Table 4.74.

Table 4.73
Student Survey Question 6 by Campus

The software covered 7 chapters. Please check all you used:	1		2		3		5		7		Total
Port Huron - 8:00	1	6%					1	6%	15	88%	17
Port Huron - 10:00			1	7%	1	7%			13	87%	15
Owosso - 10:00			1	33%					2	67%	3
Owosso - 4:00	7	39%	3	17%					8	44%	18
Muskegon - 10:00	1	17%			1	17%			4	67%	6
Flint - 6:00	1	8%					2	17%	9	75%	12
All sections	10	14%	5	7%	2	3%	3	4%	51	72%	71

Table 4.74
Student Survey Question 6 - Statistical Summary

The software covered 7 chapters. Please check all you used:	Count	Low	High	Mean	Standard Deviation
All sections	71	1	7	5.61	2.3633

Table 4.75 compares the number of chapters reported by those students who were introduced to the software in a computer lab setting and those that were introduced with a demonstration. The computer lab introduction was about one hour long. In that time period, students would have time to examine one or possibly two chapters. Several students probably only used the software during this introductory lab. Therefore, I would expect a larger number of students reporting one or two chapters used in those sections introduced in a lab, than those introduced with a demonstration. Table 4.75 supports my expectation. Interestingly, the same percentage of students reported using all seven chapters regardless of the method of introduction.

Table 4.75
Student Survey Question 6 by Introduction

The software covered 7 chapters. Please check all you used:	1		2		3		5		7		Total
Lab	8	15%	5	9%	1	2%	1	2%	38	72%	53
Demo	2	11%			1	6%	2	11%	13	72%	18

Table 4.76 compares the number of chapters reported by those students who were required or given credit by the instructor and those that were not required or given credit. As I expected, a higher percentage of those students that were required to use the program reported using all seven chapters.

Table 4.76
Student Survey Question 6 by Credit

The software covered 7 chapters. Please check all you used:	1		2		3		5		7		Total
Not Required	8	17%	5	10%	1	2%	2	4%	32	67%	48
Required/Credit	2	9%			1	4%	1	4%	19	83%	23

As mentioned before, number of hours used (question 7) as shown in Table 4.77, is also an indication of continued perceptions of use. As mentioned before, I converted the responses to this question into numeric data that could be analyzed statistically by dropping any fractional hours and using the lower value of any range. Therefore the students shown in Table 4.77 as responding with zero hours actually responded with a value greater than zero but less than one. Both of these students were in sections that were introduced to the computer instructional aid in a computer lab. Apparently they interpreted the time involved as less than one hour.

In separating results by whether or not students were introduced to the computer instructional aid in a computer lab or with a demonstration, I arrived at Table 4.78. Regardless of which method was used to introduce the program, 25% reported using it less than one hour and 25% reported using it two to three hours.

Table 4.79 compares how many hours were reported by those students who were required to use the computer instructional aid or received credit for using it with those students who were not required. Adding percentages, 33% of those not required to use the aid reported using it less than two hours compared with 10% of those required to use the aid. Looking back at Table 4.80 it can be seen that 33% of those not required to use the aid also reported using less than seven chapters. Examining the other end of Table 4.79,

20% of those required to use the aid reported using it for six hours or more compared with 18% of those not required to use the aid.

Statistical summary data for question seven regarding number of hours used is shown in Table 4.80. The students indicated as "No Answer" were not included in the calculations. The section with the largest mean number of hours, Port Huron - 10:00, was not required to use the computer instructional aid. The section with the smallest mean was the Owosso - 4:00, which also was not required to use the aid. Both classes were introduced to the aid in a computer lab. The 8:00 and 10:00 Port Huron sections also had the largest percentages of students reporting using all seven chapters (Table 4.73) and the 4:00 Owosso section had the smallest percentage of students reporting using all seven chapters.

Table 4.77
Student Survey Question 7 by Campus

How many hours?	0		1		2		3		4		5		6		7		8		9		10		Total
Port Huron-8			1	7%	3	20%	4	27%	1	7%	3	20%	2	13%	1	7%							15
Port Huron-10			1	7%	3	20%	4	27%	1	7%	1	7%			1	7%	1	7%			3	20%	15
Owosso-10					2	100																	2
Owosso-4	2	13%	8	50%	4	25%	1	6%													1	6%	16
Muskegon-10			1	17%	2	33%	1	17%	1	17%							1	17%					6
Flint-6			3	30%	2	20%	1	10%	1	10%	1	10%							1	10%	1	10%	10
All sections	2	3%	14	22%	16	25%	11	17%	4	6%	5	8%	2	3%	2	3%	2	3%	1	2%	5	8%	64

Table 4.78
Student Survey Question 7 by Introduction

How many hours?	0		1		2		3		4		5		6		7		8		9		10		Total
Lab	2	4%	10	21%	12	25%	9	19%	2	4%	4	8%	2	4%	2	4%	1	2%			4	8%	48
Demo			4	25%	4	25%	2	13%	2	13%	1	6%					1	6%	1	6%	1	6%	16

Table 4.79
Student Survey Question 7 by Credit

How many hours?	0		1		2		3		4		5		6		7		8		9		10		Total
Not Required	2	5%	12	28%	11	26%	6	14%	2	5%	2	5%			1	2%	1	2%		2%	5	12%	43
Required/Credit			2	10%	5	24%	5	24%	2	10%	3	14%	2	10%	1	5%	1	5%					21

Table 4.80
Student Survey Question 7 - Statistical Summary

How many hours total did you use the software?	No Answer	Count	Low	High	Mean	Standard Deviation
Port Huron - 8:00	6	15	1.33	7	3.92	1.7482
Port Huron - 10:00	1	15	1	10	4.90	3.2304
Owosso - 10:00	3	2	2	2	2	0
Owosso - 4:00	5	16	.675	10	1.98	2.2305
Muskegon - 10:00		6	1	8	3.54	2.3896
Flint - 6:00		10	1	10	3.95	3.3620
All sections		64	.675	10	3.57	2.7285

In response to question 1, (Table 4.81), over 73% of student responses indicated it was "Very Useful" and a total of 92% found it "Somewhat Useful" or "Very Useful." This was the first question on the student survey and was of a very general question nature. Therefore responses can not be considered to measure only initial perceptions of usefulness or continued perceptions of usefulness. Although Table 4.82 shows only three categories of response, there was a fourth on the form, "Not At All." I omitted that column here because none of the students that responded to this question on the student survey form choose that option.

Table 4.81
Student Survey Question 1 by Campus

How useful was the software?	Very Useful		Somewhat Useful		Little Bit		Total
Port Huron - 8:00	18	78%	2	9%	3	13%	23
Port Huron - 10:00	15	94%			1	6%	16
Owosso - 10:00	5	100%					5
Owosso - 4:00	8	44%	8	44%	2	11%	18
Muskegon - 10:00	7	78%	2	22%			9
Flint – 6:00	8	62%	4	31%	1	8%	13
All sections	61	73%	16	19%	7	8%	84

Finally, in order to measure perceived usefulness to students, I simply tried to identify the percentage of students who actually used the computer instructional aid

outside of class. This turned out to be more difficult to determine than I first assumed. I first tried to identify how many students had the opportunity to use the aid. 107 students received the computer instructional aid as shown in Table 4.82. An additional 29 students in the Owosso 10:00 class received the computer instructional aid. However, since I was going to use student survey forms to identify usage and so few students in that section were given the opportunity to complete a survey form (see Implementation Case Study #2), I decided to drop the class entirely from this calculation. Since the Port Huron and Owosso students used the computer instructional aid in class for about an hour, I excluded students from those sections who reported using the aid for one hour or less in determining the number of students who used the aid outside of class. Using the totals in Table 4.82, at least 46.30% of all students reported using the computer instructional aid outside of class and at least 47.22% reported using all seven chapters in the aid. One student in the Port Huron 8:00 section and two students in the Owosso 4:00 section reported using all seven chapters but reported no number of hours resulting in more students reporting using all seven chapters than reporting using the program outside of class. The above percentages indicate minimums because nine students received the program and did not take the final exam, another eleven students received the program and took the final exam but did not fill out a survey form and another eleven students received the program and filled out a survey form, but did not answer any questions of the back of the form. Since I did not receive responses from any of these students, I cannot count any of them as using the computer instructional aid even though some of them may have.

Table 4.82
Numbers of Students by Use and Campus

CAMPUS	RECEIVED PROGRAM	HOURS > 0 (QUES #7)	OUTSIDE CLASS	ALL 7 CHAPTERS
Port Huron-8:00	21	15	14	15
Port Huron-10:00	19	15	14	13
Owosso-4:00	25	16	6	8
Muskegon	18	6	6	4
Flint	25	10	10	9
TOTALS	108	68	50	51

Usefulness to Students - Actual

Although the program was not likely to make a significant difference in final exam scores (Table 4.83), given all the variables that could be of influence, I wanted to check it as an additional "selling point." I compared the final exam scores of the two Port Huron sections that received the computer instructional aid and the one Port Huron section that did not receive the computer instructional aid, the summary data in Table 4.28 was obtained. As described earlier, since the difference in the means was not in the desired direction, I decided not to conduct a t-test.

Table 4.83
Test Results by Campus

TEST RESULTS (scores in percent)	Count	Low	High	Mean	Standard Deviation
Port Huron - 8:00	21	67	100	89.00	10.2225
Port Huron - 10:00	16	48	99	83.94	14.9330
Owosso - 10:00	29	64	100	90.21	8.1688
Owosso - 4:00	24	65	100	91.67	9.2762
Muskegon - 10:00	12	51	99	79.75	16.1083
Flint - 6:00	24	62	100	87.63	10.9735
All sections	126	48	100	88.00	11.4807

As previously mentioned, I also examined the difference between an individual's pretest score and final exam score. I did perform a t-test comparing the two Port Huron sections that received the computer instructional aid with the one section that

did not (Table 4.29). I could not show a statistically significant difference between the treatment group's improvement from the pretest to the final and the control group's improvement from the pretest to the final.

I was not surprised that none of my comparisons showed a statistically significant difference. As discussed in chapter 2, Clark (1985) and others have described in detail the difficulty in showing statistically significant increases in measurement tools, such as this test, as a result of using computer assisted instruction and aids. The large number of variables that can affect the scores on tests make it difficult to show an improvement as the result of a change in only one of those variables. For example, the final exam score ceiling of 100% limits the amount of improvement that can be shown as a result of using the computer instructional aid.

Another question possibly useful in persuading others to use the program was "Would there be a statistically significant relationship between the amount of student use of the computer program and their scores on the final exam?" Pearson's product-moment correlation coefficient was used to compare final exam scores with the number of chapters used and the hours used as reported on the student survey. The relationships were weak: .22 between final exam scores and number of chapters used, .12 between final exam scores and number of hours used.

An indirect indicator of actual student usefulness is where the students used the instructional aid. Students did use the computer instructional aid in many places although the most common were definitely at home and a Baker computer lab (Table 4.84). There did seem to be substantial differences in where students used the computer instructional aid amongst sections. The sections reporting the lowest percentages of students using the

computer instructional aid in a Baker lab were the Muskegon (17%) and Flint (8%) sections. These were also the sections reporting the highest home usage (67% and 83% respectively). This might be related to the fact that these were the only two sections not introduced to the aid in a computer lab. This difference becomes even more pronounced when the responses from question five are summarized by how the students were introduced to the computer instructional aid (Table 4.85).

Table 4.84
Student Survey Question 5 by Campus

Where did you use the program?	Home		Friend or Relative		Work		Baker Lab		Other		Total
Port Huron - 8:00	9	38%	2	8%			13	54%			24
Port Huron - 10:00	10	38%	2	8%	2	8%	11	42%	1	4%	26
Owosso - 10:00	2	33%			1	17%	2	33%	1	17%	6
Owosso - 4:00	4	21%	1	5%	1	5%	13	68%			19
Muskegon - 10:00	4	67%	1	17%			1	17%			6
Flint - 6:00	10	83%			1	8%	1	8%			12
All sections	39	42%	6	6%	5	5%	41	44%	2	2%	93

Table 4.85
Student Survey Question 5 by Introduction

Where did you use the program?	Home		Friend or Relative		Work		Baker Lab		Other		Total
Lab	14	22%	5	7%	4	5%	39	52%	2	3%	75
Demo	14	78%	1	6%	1	6%	2	11%			18

Usefulness to Instructors - Perceived (Influence)

One indication of usefulness, as perceived by instructors, would be, "Would teachers influence their students to use it?" After seeing the program, all four instructors encouraged their students to use the program. The Port Huron instructor required her 8:00 classes to use the program by giving credit for doing so. (The 10:00 class was not required or given credit as a control.) The Muskegon instructor gave her students extra credit for using the program.

Another major indicator of perceived usefulness to instructors is their responses to the "post" interview questions (Table 4.30). As described before, all four instructors responded very favorably and indicated they would like to use the computer instructional aid again. All of the instructors were contacted in the summer of 1999 and asked if they were still using the aid. Unfortunately the Port Huron campus was no longer teaching that course although Joyce told me she would have continued to use it if she was still teaching the course.

As mentioned before, Cynthia, on the Flint campus, said "I did use it Winter 99 term. About a half dozen students took advantage of it and did well." She also passed the program on to another instructor who was teaching the course for the first time.

Judy on the Owosso campus also offered it to her students in winter and spring quarters and each time five to ten students used it. She noted, "The students who took it and used it needed the help or, at least, thought they needed the help." Judy felt that "anything that presents the information in a different way has got to be a help."

I had a little trouble contacting Tina on the Muskegon campus. When I was finally able to reach her, she told me she had offered the program and a few students had taken advantage of it. The students who used it were pleased with the computer instructional aid and felt it helped. Tina then told me she would no longer be teaching there in the fall.

In fall of 1999 I sent an email message to the CIS academic dean on each of the seven major campuses. These are the administrators responsible for the INF111 course. I reminded them of the computer instructional aid I had mentioned at an earlier system CIS committee meeting. I gave them my Internet web page address, from which the program

could be downloaded and encouraged them to pass the information on to their INF111 instructors and I would be interested in hearing from anyone using it. I did not receive any replies.

Then, in February of 2000, I started corresponding via email with a Muskegon instructor that I have never met. I had asked for some help regarding an advanced computer programming course we were both teaching and he sent me some information. Then apparently he recognized my name and sent me a second message that said, "Hey, you are the guy who did the WONDERFUL inf111 student basic test kit! We LOVED your program and are indebted to you so I am even more motivated to help. I suggested that students write you thank you letters."

I thanked him for the compliment and mentioned that it was part of a Ph.D. I was working on. I also mentioned that I was pleased that apparently word was getting around about the availability of the computer instructional aid. He replied that he was also working on a dissertation and that, "we both could probably use better marketing."

Usefulness to Instructors - Integration (Ease of Use)

Students and instructors were confronted with many different situations. Joyce in Port Huron taught in a computer room and students were introduced to the computer instructional aid there. Judy in Owosso taught in a regular classroom but her students were introduced to the computer instructional aid in a computer lab. Tina in Muskegon and Cynthia in Flint taught in a regular classroom and her students were introduced to the computer instructional aid in that classroom with a demonstration. In spite of all of these differences, high percentages (87% to 100%) of all sections responded "Very Useful" or

"Somewhat useful" to question number one (Table 4.81). In all of these different types of situations, the computer instructional aid was introduced successfully.

One of the most important indicators of a teacher's actual usefulness of the computer instructional aid would be is the teacher would wish to continue to use the program in later classes. As mentioned before, two of the four teachers have since continued to make it available to their students with no outside encouragement.

Maintenance

The two primary maintenance concerns were the ability to fix minor bugs and to make changes to the test bank. I made several changes to the program in June and September of 1998 after pilot testing as described earlier. In addition, the Owosso and Port Huron classes, which were held September 21 to October 23, 1998, found a bug and two errors in the test bank. The programming error or bug occasionally prevented the student from going to a second chapter after finishing the first chapter. Instructing the students to exit and restart the program following the written instructions given them temporarily solved this. Although the program logic was complicated enough that it took a several hours to track down this error, it was relatively easy to fix once it was found.

The errors in the test bank involved one question with the wrong choice marked as the correct answer, and one question with no choice marked as the correct answer. Correcting the wrong choice was extremely easy. It simply involved changing the location of the special character, asterisk, used to identify the correct answer in the test bank.

The question with no correct answer marked was a little trickier since the program used the correct answer to the previous question as the correct answer for this one. Since the order of the questions was randomized, this meant that each time the program was run, a different choice might be indicated as being correct. This was confusing to both the students and myself until I figured out that the cause. Indicating the correct answer was then extremely easy. These corrections were easily made before the program was distributed to the Flint and Muskegon classes which were held October 26 to December 4, 1998.

CHAPTER FIVE

DISCUSSION AND CONCLUSIONS

Synopsis Of Problem And Purpose Of Study

The purpose of this study was to examine the development and diffusion of a computer instructional aid by an insider. I did this by documenting the development and assessment of an instructional aid. In this way, I described how I, as an inside instructional developer, created, developed and integrated technological change into an existing multi-section college course so that it is would be seen as useful by students and other teachers.

Administrators, educators, parents and students are continually seeking new computer-based uses of technology. As computers continually become more common in educational institutions, the work place and homes, increasingly people are seeking educational software. Unfortunately, creating this software requires effort, time and expertise. General-purpose programs are usually developed by large software companies who can spread the cost to all those who might buy their products. Developing customized programs and computer instructional aids, especially for a narrow specific topic, must be done on a much smaller scale where the amount of effort and time must often be kept to a minimum. When these costs can be justified, the usual method is for the organization or client to contract with an outside technology expert who can provide the necessary expertise to create the aid. That outside technology expert must then consult with a subject matter expert to obtain the information needed to create the

computer based instruction. The subject matter expert is usually a member of the client organization that has contracted the outside technology expert.

One alternative to the traditional approach would be for an insider, a member of the organization, to be the developer. Of course, this would require the insider to be both a technical expert and a subject matter expert. A major advantage would be the elimination of all of the usual communication and coordination between the technical expert and the subject matter expert. A major disadvantage would be finding an insider who possessed the necessary technical expertise and subject matter knowledge. What other advantages and disadvantages would present themselves? What would it be like if the insider were the developer compared with the typical outside developer situation? These are questions this study tried to examine.

Discussion of Results

As discussed in chapter 2, there have been many attempts to identify the major stages in the development of a computer system (SDLC), instruction (ADDIE and Dick and Carey), the use of instructional technology, and the diffusion of innovation (Havelock and Rogers). These lists try to describe and prescribe the major stages in the overall process of creating a system from the viewpoint of a systems analyst or developer who is concerned with the entire system. In evaluating the results of this study I will follow the major stages common to all of these process descriptions. Actually, I will begin with a discussion of the preliminary development stage, which is not included in these process descriptions, and then I will examine the stages of analysis, design, development, implementation and evaluation.

Preliminary Development

A preliminary development stage is not in any of the lists described in the instructional design, educational technology, SDLC or diffusion of innovation theories (see chapter 2). The instructional design model typically begins with an analysis of the problem and situation. The educational technology model also begins with analysis of the problem. Some versions of the technological model, SDLC, do separate the definition of the problem from its analysis referring to this stage as preliminary investigation or planning. The innovation diffusion model also separates the recognition of the problem or need from research. However none of these models emphasize the prior experiences or status of the participants in the development of a system.

Traditionally, from an outside developer's viewpoint, this preliminary stage consists of the prior experiences and skills the developer brings to the situation. The outside developer perceives his or her position as being quite objective and tends to overlook the fact that his or her actions and decisions are usually based on events that occurred prior to the beginning of a project. Since the outside developer typically comes to the project, its beginning is seen as when the developer focused his or her attention on the project.

The outsider may have had extensive experience with similar problems, computers, and instructional systems. If the outsider worked on many instructional development projects before, then he or she has also had substantial experience with the development process and prior programs, courses and instructional aids that were

developed. But the outsider probably has little experience with the specific organization and environment involved with the new system that is to be developed.

In contrast, the insider is more enmeshed in the situation and environment. Sometimes the insider does not see the beginning of the project but it grows or evolves out of the prior situation and experience. The insider has substantial experience with the organization and environment but may have limited experience with the processes of instructional design, development and use of technology, and the diffusion of innovation.

In the present case, during the development and implementation of the computer instructional aid, I gave little thought to things that had happened prior to January 1998 when I seriously started working on the aid. However, while trying to document the process (see chapter 4) I continually became aware of the effect of events prior to January 1998. Although written "from scratch," the first draft or "Try" was amazingly similar to the program I had written ten years earlier in January of 1988. It used a loop to present a question to the user, accept an answer from the user, compare that answer with the correct answer, and provide feedback to the user. While doing the documenting process I managed to find a copy of that program in my files (Appendix A). The basic loop structure was the same in both programs even though the original program design was only in my memory when I started working on this computer instructional aid. In trying to find a copy of that original program I also stumbled on a version I had used in a graduate class in 1990 that I had completely forgotten (Appendix C).

Finding and examining these early versions made me realize how much I owed them. I used their looping design and gained confidence to begin the design of the computer instructional aid developed in this project. The basic design made sense to me

and helped guide my early efforts. These early versions, even the one I didn't specifically remember, provided me with the knowledge that the program I wished to create was possible and not beyond my capabilities.

The construction of an instructional aid and a computer program typically starts with an analysis of the problem and identifying the requirements for its solution. Seldom do we consider that the prior experiences of the developer or developers have a substantial influence on the solution that is eventually created. Although not a rule, many different programming textbooks, even those teaching different languages, often use the same types of programs. The idea is that the student, by learning and saving these assignments, can use these typical designs and programming techniques to create similar programs on the job. For example, most programming language textbooks that deal with data files, start with a simple program that reads and prints the data in the file. This example is used not just because it is a simple program but also because it is one of the most common programs used with data files.

Similarly, the more familiar one is with the design of similar computer instructional aids, the easier it will be to design a new aid. Thus, the ideal inside developer should obtain as much knowledge and experience as possible in the areas of instructional design; and uses of technology, especially in education.

In this case, I had taken many graduate courses in instructional design and educational technology while working on my masters and Ph.D. As an experienced instructor, I was expected and volunteered to design several new courses and rewrite many existing courses. I had worked with and taught many courses involving computer applications and programming. I also often had the opportunity to teach most courses

more than once. This experience helped me identify potential problem areas. This also allowed me try to troubleshoot those problems by trying to avoid or fix those problem areas the next time I taught that course. Organizations and individuals need to recognize the importance of identifying and resolving educational problems and provide instructors with time and training to be able to troubleshoot.

The insider's knowledge of the course objectives, content, presentation methods, typical practice, and tests can be used to advantage in the instructional design of a useful component for that course.

In addition to possessing technical knowledge and skill, the ideal insider has the advantage of knowing many people in the organization, the typical processes used in the organization for making decisions, the availability of technology, the subject matter and the objectives of the organization and the specific learning experience, lesson or course involved. Thus, the ideal designer should be consciously aware of the diffusion of innovation. This knowledge, which should have been acquired prior to development, makes it relatively easy for the insider to determine the likelihood of the instructional aid being accepted by the organization and whether or not the project is worth the time and effort that must be invested.

In conclusion, my experience has shown that the individual's experience and background in the organization, instructional design, technology and the diffusion of innovation will have a strong influence on the inside developer's success.

Analysis

The analysis process is basically the same in instructional design, education and technology, and diffusion of innovation. If the developer of a system is an outsider, one of the first things he or she must do is specifically identify the problem and the requirements that must be met in order to solve that problem Sometimes this analysis involves identifying the history of the situation: what events led up to the present problem and might have caused the problem. Sometimes this investigation involves examining and analyzing the existing system, if one exists.

Traditionally, theorists make assumptions that when a problem exists and an instructional system does not exist, a new instruction system needs is to be developed. In any case, the outsider tries to bring a fresh objective viewpoint to the situation and arrive at a description of the problem and the requirements necessary to solve it.

In the case of the aid I developed, the instructional system existed but had some weaknesses. Being an insider, I was already familiar with the desired course outcomes, the course content, the time limitations imposed on the instructor, the physical limitations placed on instructors (rooms and computer availability) and the measurement tool, the common final exam. I also knew the organization, many of the key administrators, and some of the instructors. I had taught this course many times so I was familiar with the typical student and the wide range of students in this course. This knowledge came out of my experiences prior to the development of this computer instructional aid.

As an insider, I did not have to spend the time researching and interviewing people to get to know the situation and the problem. I had already done that as part of my responsibilities as an instructor. However, an outsider, being more objective, might have

been able to identify the problem and requirements more quickly than an insider once he or she had acquired knowledge of the situation and problem. Perhaps an outsider might have focused on a completely different problem. Perhaps an outsider would have chosen to revamp the entire course rather than just a portion and whether a real problem and need for a complete change was called for.

I cannot state how long in hours it took me, an insider, to identify the problem and requirements since they grew slowly over a long period of time during the prior development stage. During the two-year period from the start of the development of a common final exam to the start of this project, I was continually evaluating the instructional strategies and tools being used to prepare students for that common final. My knowledge of instructional design helped me identify a problem of inappropriate or insufficient practice.

In fact, it is difficult for me to separate this stage from the prior development stage and the following design stage. Once I formally began the project, the analysis stage only took me an hour or two. I simply identified my perception of the problem and the requirements for its solution. It is important to occasionally evaluate one's experiences, to reflect on what recurring problems one has to deal with and to identify what would be necessary to solve or reduce these problems. It was such reflection that let to my identifying a problem and my attempts to alleviate it. This reflection can help the insider take advantage of the extensive experience he or she has that the outsider does not have and can help increase the insider's objectivity.

Then I presented this description and requirements to two colleagues, Lori in Owosso and Joyce in Port Huron, for confirmation. In retrospect I am very pleased that I

consulted with my colleagues. The major disadvantage of the insider is possible lack of objectivity. Comparing viewpoints helped insure I viewed the problem and requirements objectively. Without such comparisons it would have been easy for me to overlook or over value the requirements of the problem. These contacts were also necessary to help me design and develop a computer instructional aid that I could later sell to them and assist in the diffusion of the computer instructional aid. The processes of instructional design, development of technology, diffusion of innovation, merged.

Ideally, as an insider performs his or her duties, he or she should make an effort to spend time (1) reflecting and evaluating his or her experiences and (2) discussing those experiences with colleagues or others that might have similar experiences. Reflecting on experiences can help the instructor identify weaknesses and causes of problems. The discussions with colleagues needs to include those peers who have experience and are knowledgeable of the situation and who may be influential in the later diffusion of the instructional aid. If possible, the insider should have some job aids to help troubleshoot weaknesses in instruction, especially if he or she is not well versed in instructional design. At the minimum, the inside developer should know about the consistency of key elements in an instructional system to be able to detect inconsistencies and omissions.

Design

Typically, the design stage involves making decisions about the new system including its components and the relationship between them. Design is the second "D" in the ADDIE instructional design process model and a stage in almost every SDLC process model. Research (basic and applied) is the second phase of Rogers' Innovation–

Development Process, a well-known innovation diffusion model. This research phase relates more to scientific investigations of new technology. However, design and decision-making would be incorporated in this phase of the innovation diffusion model.

Traditionally, in the design stage the outsider analyzes the information that he or she obtained in trying to identify the problem and the requirements for its solution. Using this analysis, he or she tries to design a solution that will meet these requirements. During this process the outsider typically focuses on the problem and its requirements. Recall that in design models the assumption is made that a whole instructional system will be created. Since the outside developer would be expected to have extensive knowledge of these models and have had good success using them, the outside designer could be expected to be inclined to recommend development of a completely new system.

The outsider has the advantage of experience focused on designing and creating many and varied systems. The outsider is also more likely to be aware of new approaches and new technologies that might be used to solve the problem. The outsider, because of his or her varied experience, may know of possible solutions of similar problems in completely different fields. This is likely to lead to relatively innovative solutions.

At the end of the design phase, to insure successful diffusion, the outsider must present and obtain approval for the solution he or she has designed. The outsider typically has a contract or some other type of agreement, usually with the administration. In any case, the outsider must obtain approval from whomever that agreement is with, before proceeding. If the outsider is wise he or she will check with likely users of the solution to insure later diffusion.

In contrast, the insider focuses as much on the situation and the environment as he or she focuses on the problem itself. In designing a solution, the insider is continually asking, "What will work here." The insider's advantage is knowledge of the problem and its relation to the existing situation. In terms of design, this means the insider may better know how to integrate the proposed solution into existing systems. The insider would have a better understanding of what is practical. That is, how much effort can be expected from those primarily involved, instructors and students. The insider may have a better understanding of the importance of various factors and individuals in the successful acceptance of this system so he or she can design around those problems. However, the insider may be overly constrained by prior experiences and limitations, which perhaps, could be changed. This may limit the degree of innovation.

In this case the insider saw the solution as the enhancement of one part of the existing instructional system. This is a departure from the usual approach of creating a whole system. However when changing one aspect of an instructional system the other components are likely to be affected. Thus the insider needed to know all of the instructional components and how they are related to each other as well as the way these components were used by the faculty.

In this situation, I also knew that student use would most directly be affected by the introduction of the product. The product needed to be sold to the user, the student. Because this form of instructional marketing was important, I arranged to present the computer instructional aid to the students personally. In addition, student use might also be strongly affected by the instructor's attitude. Minimally, I needed the instructor to be open-minded. Preferably, I wanted the instructor's support and recommendation.

Therefore, I designed the computer instructional aid to require little instructor time and effort. I needed to design the aid so it could be used completely independent of classroom activities. I needed to avoid the impression that this was one more thing the instructor had to do or work into limited classroom time. Generally speaking, I tried to design this aid from the instructor's viewpoint. I tried to identify the key factors, such as minimal support, repetition and feedback that would be important to an instructor. Since I had taught this class many times and consulted with two other instructors who often taught this class, this was relatively easy for me to do. However, designing the aid for instructors might have been an extra hurdle for an outsider.

In contrast to a typical outsider's concern, administrative support was not that important in this situation. Within the classroom setting, the instructor is very independent of administrative management and interference. In the long run, the instructor in a small college or community college that relies on many part-time instructors cannot openly flaunt administrative policies for he or she will not be hired back for another quarter. However, within certain limitations, the instructor is given a large amount of academic freedom in meeting the course outcomes. Therefore, I did not seek administrative input during the design phase. I felt this input would be unnecessary and an inefficient use of my time. However, while I managed to avoid the need for administrative approval of my design, I did need to obtain approval of my design from my first key group of users: instructors. And, being an insider, I knew whom the key instructors would be, those who would make a difference in diffusion. Obtaining input and approval from them was one of the major goals of my pilot study.

So, in designing a system, the outsider has more experience designing systems while the insider has more experience working within the organization where the system will be used. To compensate, an outsider needs to focus more on the client organization while the insider needs to work on increasing his or her instructional design, technology, and diffusion experiences. In this stage, Havelock's idea of a combined inside-outside team would have substantial advantages. However, the time to understand each other and get to a working arrangement is an extra expense.

The inside developer has the advantage of knowledge of the course and thus is better able to match practice to test and to objectives to insure instructional consistency. Of course, this requires the inside developer to know instructional design. Inside knowledge also helped me identify the solution requirements during analysis stage, especially those that used my knowledge of the customers or users of the computer instructional aid and the need to make it viable and useful.

The insider must be very aware of his or her limitations. The insider must be very careful that he or she does not take on a project that exceeds his or her abilities. If a project does push or exceed the insider's skills and experience, he or she could seek additional study and experience, if completion time is not a problem. Another alternative would be for the insider to obtain outside assistance from someone with more knowledge and experience in the insider's weak areas. The insider must also be aware of his or her time limitations. The insider has his or her regular duties to perform and cannot ignore those responsibilities while working on a project. Unless the insider can arrange some reduction in workload, he or she will have to find extra time for this project.

In summary, the ideal design phase for the insider would be where the advantages of his or her knowledge of the organization would be more important than the disadvantage of his or her limited skill and experience in designing. This would include circumstances where acceptance is more important by specific individuals rather than everyone in the organization, where a portion of instruction is being developed which does not require complicated design such as a component change or addition rather than a complete major system development.

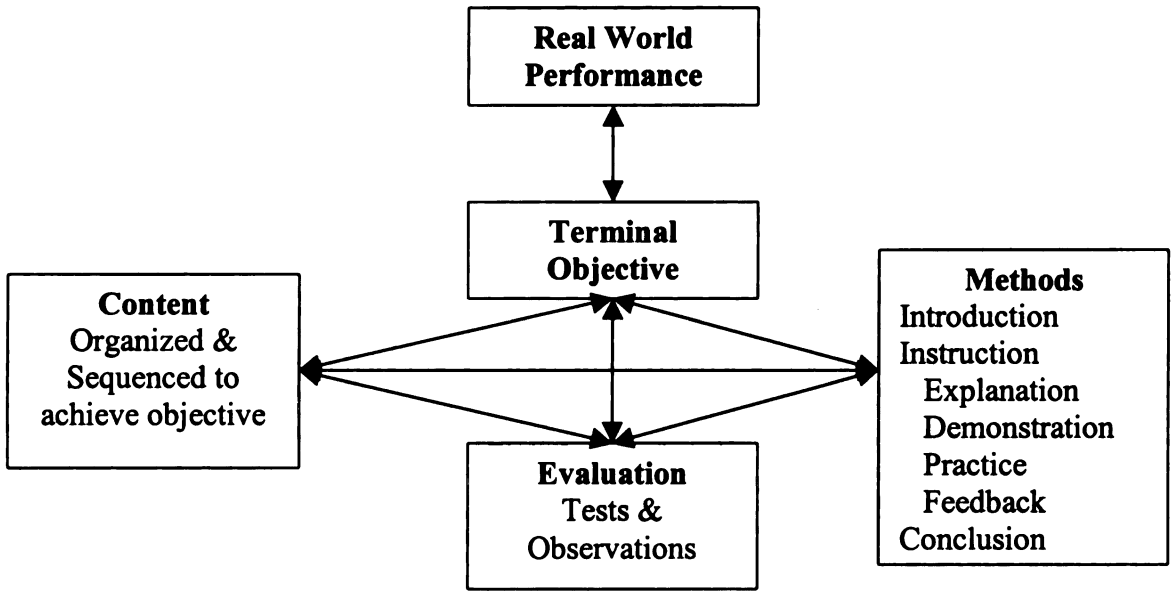

Figure 5.1. Yelon's Consistency Model

The design of the part would have to be consistent with other elements in the system. This consistency is "'the secret' of instructional design in course development and revision" (Yelon, 1996). To summarize Yelon's secret (Figure 5.1), the terminal objective, the methods including practice, the content and the evaluation should all be consistent. Plus, the terminal objective should match the desired real world performance.

The element being designed should also appear easy to use by the instructor and likely to help the students, but only use computer technology when it is an appropriate medium.

Development

The development phase in the Instructional Design process model (the second D in ADDIE) is basically the same as the development stage in the educational technology process model, the creation phase in the SDLC computer system process model and the development stage in the diffusion of innovation process model.

To the outside developer, developing the system involves obtaining any necessary hardware and software, converting any data, hiring any additional people, writing procedures for people who will need to work with the new system and training those people. During this creation process, the outsider would follow the plan made and approved in the design phase.

If the new system requires software to be created, either that task is turned over to a programmer or the developer takes on that role. Typically, the programmer's task involves several phases, called the Program Development Cycle, which are very similar to those followed by the developer:

1. Review requirements
2. Design program
3. Code program
4. Test and debug program

Two additional phases that are sometimes added are documenting the program and maintaining the program.

Sometimes the project is so large that an entire team of experienced professionals is involved. Because the outside developer and each of these professionals has extensive experience in his or her specialty, they have substantial advantages with large and

complex projects. Unfortunately, they also have the disadvantage of increased time and effort that must be involved in communication and coordination. But with extremely large projects, there often is no other choice.

In contrast, the insider tends to be more efficient if the project is small and simple. If the project is small enough that it can be completed by one individual within the desired time frame, and that individual has the necessary knowledge and time, the individual can be much more efficient than an outsider developer or a project team. The individual insider can eliminate the problems that require communication and coordination. The project can even be accomplished while doing full-time work.

In this case, in January 1998, I designed and created a computer instructional aid, documenting these actions. I made additional changes and adjustments to the program in June and September of 1998 plus one small error correction in the middle of the fall quarter in October. I used a prototyping process for the design and creation of the computer instructional aid (Figure 5.2 or Figure 3.4) which allowed me to design a small portion of the program, create, test and debug it, then design the next portion and continue.

Prototyping combines the SDLC stages of design and creation. From the programmer viewpoint, it combines the phases of design, code, and test and debug (Figure 5.2 or Figure 3.4). First of all, the programmer designs an initial portion of the program or, later, an additional portion or modification to the program. This includes determining what is required of the portion or modification. That portion is then coded, entered and tested. Then that portion is compared with the requirements for the portion or modification. If it does not match the requirements, the portion is debugged, which

involves rewriting the code, entering the changes and testing again. If the portion does match its requirements, the entire program is compared with its original requirements. If the entire program does match its requirements the program is done, otherwise another modification is designed.

Prototyping works best with small, relatively simple computer programs. If the program is complex and especially if it requires several programmers working together as a team, it is usually best to follow the standard Systems Development Life Cycle and Program Development Cycle.

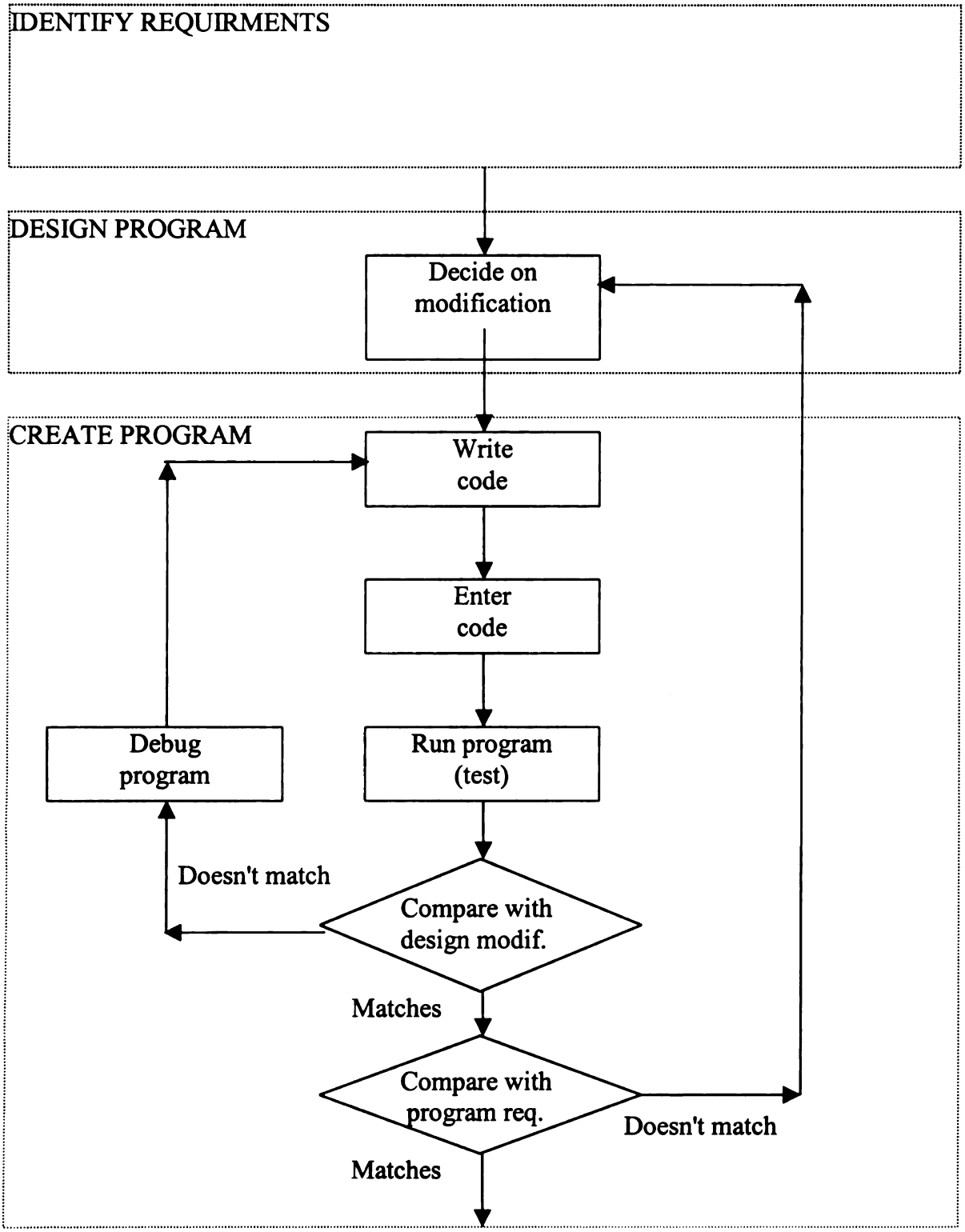

Figure 5.2 (same as Figure 3.4). Prototyping Model

In spite of my prior experiences I was surprised how difficult it was to create the computer instructional aid and eliminate all errors. If someone had asked me at the beginning of this project how many tries it would take, I would have guessed about eight or ten tries rather than the sixteen that were required. The need to allow sufficient time for programming cannot be overstated. Time limitations are a major problem for an insider. In addition to working on the project, he or she has his or her regular day-to-day responsibilities. In the case of an instructor, that means he or she has to continue to prepare for classes, teach, and do grading. In contrast, an outsider can focus full-time on the project.

Implementation

Implementation involves the placing of the system into actual use. It is the fourth stage in the standard ADDIE Instructional Design process. Implementation is also referred to in the typical instructional technology model (Keller, 1987 and Rizza, 1981) and the standard SDLC model. Rogers' diffusion of innovation model (1995) is slightly broader in that this stage is referred to as commercialization and includes "production, manufacturing, packaging, marketing, and distribution of a product" (p. 143). Rogers' diffusion model also includes another stage referred to as diffusion and adoption, which can be included in the general process of implementation.

During implementation, the outside developer must identify key individuals, distribute the new system to them, and convince them of the usefulness of the new system. If, as in this case the system primarily consists of software, distribution is not a major problem. However identifying key individuals depends completely on the

outsider's knowledge of the organization and its people. And this knowledge consists only of what the outsider has learned during the earlier phases of the project. The outsider was probably hired by one or more administrators at a higher level than the actual potential users of the system. While this might mean that the outsider has greater authority in influencing the users, success or failure depends completely on the users anticipation and perceptions of usefulness. Typically, an outsider will also have to arrange training for those users in the operation of the new system.

In contrast, the inside developer already has an extensive knowledge of the organization and its people. The insider can more easily identify those individuals who are experienced enough to appreciate the advantages of the new system and influential enough to be able to encourage its diffusion. Since the insider has more knowledge of users' ideas regarding usefulness, the insider can do a better job of designing, developing and presenting the system to meet those views of usefulness. Since the insider is designing and developing the system with the user in mind, the user training requirements will probably be minimal. Additionally, the insider probably has the advantage of being trusted by users, especially if he or she is also a user.

In spring and summer of 1998, I pilot tested this instructional aid with three small groups of students. I then contacted four instructors and during the fall of 1998, interviewed those instructors at the beginning of the course, met with six of their classes and distributed the aid, returned to the classes at final exam time to administer a survey form (questionnaire), and interviewed the instructors at the end of the course. I then reviewed, summarized, analyzed, and reflected on the data that I had collected.

I underestimated the difficulty in distributing the program and student survey form. During the pilot phase, the Lori and Joyce distributed the computer instructional aid themselves. During that time I imagined that during the implementation phase, fall of 1998, I could just send copies to all of the instructors involved and they would make copies, distribute, and then later collect them for me. As I contacted instructors and made arrangements at the end of the summer of 1998, I realized that to insure that distribution was done and students were encouraged to use the aid, I needed to make and distribute the copies myself. It was one thing to ask an instructor to give me some class time, it was another to ask an instructor to take care of some of the paperwork for me. After my initial contacts with instructors I realized that requesting that much of them might be too much of a favor to ask.

As noted earlier, I also had some problems with diffusion of the program. Two of the instructors I worked with left the college. One of these instructors was full-time and the other part-time. The Port Huron campus dropped the course completely. Changes in faculty, especially part-time faculty, and curriculum are fairly common events. It is important during the planning and design of a project that these possibilities are taken into consideration.

For ideal implementation, the major need of the inside developer is adequate time to produce, market and distribute the instructional product. It is the insider's knowledge of the organization and its people that is the most critical element in this stage. So the insider needs to be sure and base his or her decisions and actions on that knowledge. Specifically, this includes decisions related to (1) size of element and degree of change; (2) ease of use, in regards to time and effort, for both instructors and students; (3) and to

whom and how to market. These decisions can benefit from knowledge regarding (1) amount of change that will be tolerated; (2) ability of instructors and students and amount of effort that can be expected; and (3) amount of enthusiasm, or at least interest, in change of different potential participants.

Evaluation and Maintenance

Evaluation is the fifth phase of the ADDIE process model for Instructional Design. Both formative and summative evaluation are key components of the typical instructional technology process model (Keller, 1987 and Rizza, 1981). SDLC process models typically identify testing and debugging at the end of the development stage, which is a type of formative evaluation. Less than 30% of the SDLC models surveyed do include a summative or post-implementation evaluation or final review (Table 2.1). Even those SDLC models that do not specifically mention evaluation do include some type of maintenance and support of operations. Roger's diffusion of innovation model does not include implementation. The final stage in the diffusion model refers to the consequences of the change that can be seen in the changes to the individual or organization. This consequences stage does include the identification of desirable and undesirable consequences, which is a type of evaluation.

An outside developer traditionally performs only one summative evaluation at the conclusion of the project. This evaluation may be related to official completion of the outsider's contract. As a result the outside developer would have a strong motivation to show significant measurable learning improvement. In any case, the outside developer's involvement usually ends with this evaluation. Rarely, the outsider may perform a long-

term follow-up contact, but usually only if this was a part of the original arrangement with the developer. Unless errors or bugs are later found in the system, there is no further contact between the organization and the outside developer.

During the pilot testing of the program in spring of 1998 and the actual implementation in fall of 1999 I did perform minor maintenance on the program to fix a few errors and simplify the menu for users (see chapter 3). During the winter and spring of 1999 I compiled data from my interviews and surveys. I was pleased that all instructors and almost all students who responded to the survey, found the computer instructional aid useful. As noted before, I did have difficulty identifying "non-users" for statistical evaluation purposes. I did make some minor typographical corrections to the program at that point and uploaded it to my personal web page so students and other faculty could easily obtain it over the Internet. In the winter of 1999 and prior to the fall of 1999 I emailed the academic dean on each campus who was responsible for the INF courses. I informed the deans of this program, gave instructions for downloading it from my web site and encouraged them to share this with their INF111 instructors.

During evaluation, the inside developer is more "invested" in the organization and the newly created system. This means that the insider may have more difficulty evaluating the system objectively. Therefore, it is especially important for the insider to obtain evaluations from all interested parties and to give their responses careful consideration. The ideal insider also needs to carefully pre-determine the evaluation methods that will be used during the analysis and design stages.

Unfortunately, the insider, working on a small project, is not likely to be able to produce measurable improvement. In fact, measurement may not even be possible

because of all the factors involved in the educational experience and the minor role of the small project. That's why use is so important as a measurement tool. In this study I defined use as the actual student use of the computer instructional aid outside of class as reported by students on the student survey. I also used the perceptions of use by students and instructors as measured by their responses in the survey and interviews respectively, when asked if they found the aid useful.

After evaluation an outside developer typically leaves but an inside developer will probably remain in the organization. This means it will be easier for the insider to continue to evaluate and maintain the product. The insider needs to consider the need for adequate time for this continued support and maintenance. Finally, the insider needs to know when to quit and let the innovation continue or die based solely on usefulness.

Summary

To summarize, the outside developer has several substantial advantages. He or she probably has greater experience in design and development compared with the inside developer. The outside developer is usually able to focus on the problem and its solution full-time with no other occupational requirements. The outsider also is more likely to have substantial authority from the administrators who hired him or her.

The inside developer has several key advantages also, especially in the areas of prior knowledge. The insider usually has more knowledge of the subject matter and objectives of the course. The insider has more knowledge of the system and the problem including the class situation, the students, assessment devices such as tests and final exams, and expectations of students, instructors, and administrators. The insider also has

a greater knowledge of the organization and its people, specifically the administrators and other instructors or colleagues.

The key weaknesses of the insider are (1) the limited time and effort available and (2) the possible limited prior knowledge and experience in the areas of instructional design, technology, and diffusion of innovation. Administrators can adjust for these weaknesses by (1) finding ways to provide instructors with additional available time; and (2) providing training in the above areas.

When is inside development appropriate? One of the major considerations is the size of the project. If the project is small, especially only a part of a larger course, such that it might be possible to be completed by only one individual, an inside developer has many substantial advantages. If, however, the project is large, such as the complete development of a course, the outside developer's ability to spend more time and, perhaps, use a complete development team, gives the outsider the overall advantage.

Conclusions

Interpretations

1. One Inside Computer Developer Can Create Successful Technological Aids.

This was a personal belief of mine prior to this study. However, every typical instructional design model, educational technology model, SDLC model and diffusion of innovation model, seemed to focus on large complicated projects involving entire courses. Typically, a person working full-time, such as an outside developer, or maybe even a team would be needed for projects of this size and complexity. In this case, I was

successful because the project was small and also because I was knowledgeable in several important areas.

2. Creating a Useful Computer Instructional Aid Requires a Fairly Substantial Amount of Knowledge in Several Different Areas: Instructional Design, Educational Technology, SDLC Including Computers and Programming, Diffusion of Innovation Plus the Insider Knowledge Discussed Earlier.

Prior to this study I believed that only a little computer and programming knowledge and a small amount of insider knowledge was needed to be successful. In recording and analyzing the data in this study, I learned that much of my success was based on knowledge that I tended to undervalue.

a. An inside developer needs a solid knowledge of instructional design.

This is especially important when the product being developed is only a part of an existing course. For example, one of the key principles, even "secrets," of instructional design is consistency. The instructor should "coordinate the real-world performance goal with your test, objective, instructional method, and content" (Yelon, 1996). Throughout development, I had to continually check to be sure what I was creating would be consistent with the other parts of the course. This knowledge of design plus my "insider" knowledge of the course, its content, its tests, and typical presentation methods, was priceless. The course knowledge made it relatively easy for me to be sure the product was consistent.

b. An inside developer also needs to know a great deal about technology.

I needed to know what computers could do well and what they could not do so well. As a sole developer, I needed to know quite a few details about the computer hardware that would be available, about computer programming, and about testing and debugging a computer program. Without these skills, which I had obtained prior to the beginning of this project, it would have been difficult, if not impossible, for me to create a satisfactory computer instructional aid. A developer also needs to know the advantages and disadvantages of using technology in instruction and when it is appropriate. This requires a thorough knowledge of both technology and instruction.

c. An inside developer must understand the process of diffusion of innovation.

While I was developing a computer instructional aid for my students I was also constantly aware that, if developed properly, other instructors teaching the same course could easily use it. Since this was a multi-section course, the potential number of students who could use this aid was quite large and made it easier to justify the large amount of time and effort required for its development. I was aware that to be diffused to other instructors and their students, they would need to see this aid as potentially useful. This fact was an important factor at each step of development.

d. An inside developer needs a thorough knowledge of the course plus the organization and its people.

As discussed earlier, this is the area where the inside developer has a clear advantage. This knowledge is especially important when the product or aid being developed is only a part of an existing course. The developer must understand the course's subject matter, objectives, and other situational factors in order to create a useful

product. This knowledge also has a direct effect on the diffusion of the product developed.

3. In Order to be Successfully Diffused, the Need for Eventual Acceptance by Users Must be a Key Factor in the Design and Creation of the Product or Aid.

Throughout development, the developer must be constantly aware that one of the most important features of the product or aid is the perceived usefulness by the users. In this case, the users were the other instructors and their students. On the one hand, solid evidence of usefulness is sometimes difficulty to obtain. On the other hand, only the appearance of usefulness is necessary to get potential users to try the product or aid. The true measure of usefulness can be seen in how many of those potential users continue to use the product or aid after they have tried it.

4. A Substantial Amount of Time is Required for Development of Even a Simple Computer Instructional Aid Based on Prior Projects.

I estimate that I spent a total of 60 to 70 hours analyzing, designing, developing and testing the aid from the beginning of the project in January of 1998 until its final implementation and testing in the September of 1998. Since this aid was based on an earlier program I had written and existing test bank data files, I originally estimated that those stages would take only about 20 hours. The implementation stage also took almost three times more time than I had originally expected since I had hoped to simply send the aid to the instructors involved and let them introduce and distribute it. I had hoped that the evaluation would be a simple comparison of those who used the aid with those who

did not. As noted earlier, I was not satisfied with my ability to do this and therefore had to do a much more detailed analysis of perceived usefulness. At the beginning of this project I knew it was very easy to underestimate the amount of time that would be needed. However, I found that general perceptions of problems are no substitute for specific knowledge and experience.

Limitations and Strengths

My results are based on my experiences working on this project. This project involved a computer instructional aid that I, individually, developed for use in a college course to help students practice for a common final exam. There was the potential for substantial bias relative to my personal experience, viewpoints, and situation. Therefore, this discussion can only relate to a similar situation.

As noted earlier, I have had an extensive background in instructional design. In addition to many graduate courses, I have designed several new computer courses, and written or rewritten course objectives and outcomes for innumerable courses. This has occurred over 20 years of teaching experience in this environment, a post-secondary institution with a strong career orientation. I have had over 18 years of experience working with different types of computer hardware and software with an emphasis on computer programming and applications. During this time at this institution, I have also been involved in the diffusion of several substantial innovations including major changes in hardware, software, new applications and the introduction of an entirely new medium, the Internet. I have also accumulated substantial insider knowledge during this time period at this institution. I have taught at two of the three largest campuses in this system.

I know almost all of the full-time instructors in the Computer Information Systems (CIS) department throughout the Baker College System and many of the part-time instructors on the Owosso campus where I teach. I would consider myself a respected and influential CIS instructor in the system. Over the years I have also developed positive working relationships with many of the administrators throughout the system, especially within the CIS department.

As I have noted in this study, all of these personal factors influenced this study. Someone else, in a different situation, with different experiences is likely to have completely different results. In addition, my involvement in the development and evaluation of the computer instructional aid constitutes a major bias. I attempted to compensate for this bias by soliciting the opinions and viewpoints of colleagues and by maintaining as much objectivity, as possible, in the evaluation of the results. However, I must acknowledge that many of these results are completely subjective.

As mentioned before, this project involved a computer instructional aid developed for use in a multi-section college course emphasizing computer terminology and definitions to help students practice for a common final exam consisting of 81 multiple-choice questions. These results cannot be generalized to other types of courses, other types of students, other types of final exams or other types of institutions. This aid used a drill and practice strategy to assist students in learning specific facts. It is always important to carefully analyze the subject matter being taught and to choose an appropriate strategy and technology. This study did not examine that analysis or the decision to use that strategy and technology and the results may not relate to other types of subject matter, other teaching strategies or other technologies.

However, my background and this environment also means that this was a unique opportunity to examine the entire process of design and development by an inside developer who was very familiar with the organization, the course and its content, the people involved and was also knowledgeable in the areas of instructional design, the use of technology, and the diffusion of innovation.

The computer instructional aid developed in this study was DOS based and created in GW-BASIC. Since the time this study was begun there have been many changes in computer software, hardware, and communications. If this instructional aid was being developed today, I might decide to create it with Visual Basic or some other programming tool. I might choose to distribute it not on a floppy disk but on a CD or via the Internet. Theoretically, the results of this study cannot be generalized to other development and distribution techniques. However, nothing in the results of this study is related to the specific technology used other than the problems some students had in getting the aid to work. These problems might be increased or decreased by the use of alternative technologies.

New technology, such as a computer application, becomes truly integrated into the educational experience when (1) the features, advantages and disadvantages of its use are well known; (2) the technology is available to the instructor; and (3) the effort to use it has been minimized or at least the effort can be accurately estimated. Then the instructor simply analyzes the need by examining the objective, content, and instructional event. The instructor then applies the most appropriate and sensible tool. However in the analysis of need and application of technology, I intentionally restricted this study to several specific features of the situation. For example, I limited the objective to

improving scores on the common final and the content to the facts in that common final. I also limited my concern to the practice instructional event, to a situation that involved an individual student working outside of the classroom environment and I was strongly influenced by the availability of technology. Plus, even though I used a technology that has been around for a while and is fairly well known, a drill-and-practice program, I still underestimated the necessary effort.

If I were doing this project again to accomplish the purpose of recalling factual knowledge I would still prescribe drill and practice. However, I would give more careful consideration to alternative strategies and technologies to reduce the amount of effort. For example, I would try to increase the involvement of other instructors in the project. Input from other instructors proved much more valuable than I expected and with additional participation of other instructors I might have been able to decrease the amount of time and effort I had to put into this project to complete it.

Some technologies might have involved less effort but would have had fewer advantages. For example, flash cards could have been used to help students memorize the questions and answers. However flash cards would have eliminated the novelty factor involved with working with a computer, would not have allowed the guessing technique that the student was required to use to obtain the correct answer, and would not have simulated the test taking process as well as the computer instructional aid. There are also various inexpensive computer programs that I might have been able to use to decrease the necessary time and effort. Programs such as FlashCard Pro (http://www.memorize.com), 123 Flash Me (http://www.123flashme.com), and iMaster (http://neostarsoft.com) could be adapted to provide a computer instructional aid similar to the one constructed for this

project and would require no knowledge of programming. Unfortunately, these programs cost $10 to $30 per copy and an off-the-shelf program can never provide as much instructor control as the custom designed program created in this study.

Some technologies might have provided more features but involved substantially more effort or cost. An expert system involving artificial intelligence might have provided specific guidance and information to help a student find the correct answer when a question was answered incorrectly. Unfortunately expert systems take substantially more time to develop than typical computer programs such as the one used in this computer instructional aid. A computer instructional aid that would run over the Internet would provide the additional advantages of greater ease in changing the test bank and direct transmission of the test scores to the instructor. Certain web sites, such as www.blackboard.com, also allow the creation of tests and quizzes with no programming knowledge required. However, these alternatives require student access to the Internet in addition to access to a computer. While all computers in the Baker College System computer labs have access to the Internet, not all students who have access to computers elsewhere necessarily have access to the Internet. In addition there are several developmental problems. Use of the web site www.blackboard.com requires institutional, not just instructor, adoption and an institutional investment of $5,000 to $50,000 in hardware plus software licensing fees. Alternatively, developing one's own Internet version would require programming skill in Java to create an applet or a CGI language such as Perl to create a web server based program. All of these alternatives would allow the use of drill and practice to help the student recall factual knowledge and would

promote careful evaluation to determine if the advantages, including reduced time and effort would out weigh the disadvantages.

I would probably make changes to this project if any of the factors mentioned earlier: objectives, content, instructional event, or availability of technology; were different. If the content consisted of concepts then appropriate practice might involve an aid that presented examples and asked the student to identify the correct concept, or identify the incorrect examples of a given concept or provide examples for a given concept. If the teacher was restricted to a using a computer program to present and evaluate the students responses, all of these would have required a much more complicated program. Examples and concepts involving sounds, pictures, actions, and emotions, would be difficult if not impossible to present with a computer program. They would require at least some kind of multi-media technology such as digitized music and video involving CDs (compact disks), DVDs (Digitized Video Disks), large amounts of data storage, fast CPU speeds and powerful developmental software. Principles and procedures would probably also require much more elaborate and complicated technology to create useful simulations.

I also would consider a different technology if I was concerned with a different instructional event or physical situation. If the problem was related to the presentation of material in a classroom situation with many students, an electronic blackboard might be more appropriate. Or, if the instructor needed to provide individualized learning guidance to students outside of a classroom environment, some type of on-line communication, perhaps involving the Internet, would be a more sensible technology. However, since the focus of this study was to examine the developmental process and concentrate on the

aspects related to an inside developer, I limited this study to only one type of content, one type of situation, one instructional event and only one kind of technological application.

Importance of Study

As noted in chapter 2, Wilson (1999), Hopper (1999), Lawler et al. (1998), and Rickard (1999) have identified the importance of the instructor in the design and integration of technology into education. Noble (1998) laments that in the past, technology has been introduced "from the top down, either without any student and faculty involvement in the decision-making or despite it" (p.23). Rickard (1999) lauds the fact that today, "not only do many faculty members appear willing to pursue new forms of technology in their courses, but many more are engaging in the actual development process" (p. 42).

This study provided the opportunity to study such a situation with an interesting combination of factors. I was an inside developer. I was very knowledgeable regarding this course and the organization. I had taught this course for over 20 quarters and I had taught at this organization for almost 20 years. I knew the subject matter, the common final, the classroom situation, the typical student, many fellow instructors, most of the system administrators, and the committees and decision making processes of this organization. I also had some degree of influence because of my years of experience and relationships with many people.

I was also knowledgeable in many areas that often only an outside developer is familiar with. I had taken many graduate courses in instructional design and had been involved in the development of many new courses and making major changes to existing

courses. I also knew quite a bit about technology including educational technology. For over 15 years I had taught many computer classes including the use of computer applications, computer programming and systems theory courses. My masters degree was in educational technology and at various times I had worked with educational software products both as an instructor and as a student. I also knew a great deal about the process of diffusion of innovation. I had taken a graduate course in the subject and had been involved in various new courses and changes mentioned earlier.

The situation was also a bit unusual. The course included a specific, clearly defined common final exam that every student must take. There were a large number of instructors, sections and students who took this course each quarter. There were a large number of computers available to the students although there were many only available during certain time periods.

I learned about some new things with this project that I didn't expect. For example, I learned about the importance of prior knowledge and experience. My prior belief that one person could develop a useful computer instructional aid was supported. But I learned this requires a more substantial knowledge of instructional design, technology and diffusion than I originally estimated. I also learned, first hand, that it is very easy to underestimate the amount of time and effort required for development and diffusion. And finally, the main thing I learned is the differences between an inside developer and outside developer and what steps an insider should consider before undertaking a similar project.

If I had not undertaken this project as an inside developer, it would not have been done. There is no administrative policy, program or fund to support the creation of

computer instructional aids by outside developers. If I could have shown that there was a great enough need and benefit to justify the expense of an outside developer I might have been able to obtain approval and funding. However this would have been extremely difficult, if not impossible. To my knowledge, the institution has never funded the creation of any instructional aids or tools by an outsider.

Model to Explain Insider Development

Yelon and Sheppard (1999) proposed a model to explain the likelihood that someone would use an idea that they learned. I found the Yelon and Sheppard model very useful in helping me understand my actions, as an inside developer, and also understand what would encourage instructors and students to use the computer instructional aid that I developed. The Yelon and Sheppard cost/benefit transfer model (Figure 5.3) was based on a model developed by Slawson et al. (1994) explaining the likelihood of physicians to use medical information. Yelon and Sheppard modified and expanded the model to show that "the use of an idea on the job" is the product of the need for the idea, multiplied by the sensibility of the idea, and divided by the effort required.

To Yelon and Sheppard, use of knowledge "means more than just performing a skill partially or wholly demonstrated in training. Performers may use a total skill, strategy, or method; they may apply a rule or technique as one small part of a larger performance...." Need is defined "as the feeling by performers that they can apply an idea or skill to achieve a desired work or personal goal." Sensibility refers to "the performer's belief that an idea is credible, true, and suitable to apply." "The performer thinks the idea 'makes sense.' The idea fits the performers' experiences in the field, and

their experiences during instruction." Finally, effort is "the learners perception of the time and energy required to put the idea into use."

$$\text{Use of an idea on the job} = \frac{\text{Need} \times \text{Sensibility}}{\text{Effort}}$$

Figure 5.3. Yelon and Sheppard's Cost/Benefit Transfer Model

As an inside developer, my use of the computer instructional aid was based on my feelings that there was a great need for such an aid. Instructors, including myself, seemed to have difficulties providing adequate satisfactory practice for the final exam. The total need was quite large when I considered the many students required to take this course and the many sections taught on each campus each quarter. I perceived the sensibility as being fairly large. This aid would provide practice that closely matched the final exam and was consistent with all other instructional components. My instructional design background convinced me this aid was suitable and "made sense." One of my major conclusions was that I substantially underestimated the amount of time necessary to design and develop the computer instructional aid. Even knowing how much effort was required, I would do it again because I still believe the effort was justified because of the large amount of need. The use of an inside developer was practical in this situation because I had knowledge of the organization and I also had sufficient technical knowledge. However, from the institution's viewpoint, the need would not have been

seen as great enough to hire an outside consultant because the effort, cost in this situation, would have been too great compared with the institution's perception of the need.

The model can also be applied to the instructors. In their case, my insider knowledge helped me understand the instructors' viewpoint and what would encourage them to give the aid to their students. The instructor's goal, relative to the final exam, is to help students do well. The instructor needs the aid if he or she feels it will help him or her help students do well on the final exam. The instructor does not even need to believe that there is a problem, only that the aid will help students. To the teacher, it was important to know the format (multiple-choice questions) and the content (the test bank of the computer instructional aid) were the same as the final exam. Even though instructors were not familiar with the instructional design concept of consistency, the similarity between the aid and the final exam would encourage them to believe the aid was a sensible way to practice for the final. Sensibility or "validation of an idea can come from an outside source as well as from the performer's own experience or intuition" (p. 84) Here, the instructor's external validation came from a colleague, myself, who was familiar with the course, its content and the final exam. I was recommending the use of the aid. There was also the additional factor that the instructor was doing me a favor. Because of our relationship, my reputation and my position, it made sense for the instructor to support the use of the aid. Finally, during implementation, no effort was required of the instructor. I copied the disks and instruction sheets. I reserved the computer with the LCD projection panel and did the presentation. The instructor only needed to give me class time to present the computer instructional aid to the students. After the fall of 1998, students were not given a presentation of the computer

instructional aid. Instructors simply informed the students of the availability of the aid and made copies if asked. This did require a slightly greater effort on the part of the instructor and may have discouraged the instructors from strongly recommending the aid.

In the case of the students, use can refer to them actually using the aid to practice for the final exam. To the student, their goal is doing well on the common final exam. Students need the computer instructional aid if they feel they can use it to help them do better on the final exam. Need refers as much to feelings and perceptions as it does to any measurable benefit. If the students feel or perceive the aid will help them reach their goals, then they need it. Not all students used the aid. If a student did not feel the goal of doing well on the final exam was that important or did not believe he or she needed much help in achieving that goal, that student would be less likely to use the aid. Likewise, if the student did not believe practice or similarity between the practice and the final exam resulted in a "sensible" way to prepare for the final exam, the student was less likely to use the aid. The student's external validation came from the presenter, myself, and the instructor's endorsement of the aid. Students are encouraged, even trained, to "do what the teacher says" whether it makes sense or not.

Even during the fall of 1998, student effort was fairly substantial. Students had to find the time outside of class, had to obtain access to a computer and had to run the program. This last was especially challenging since these were beginning level computer students. Many of these students had never had a computer course before and some had never worked with a computer before.

In this study, use was high during the implementation phase but declined sharply during later quarters. This coincided with the fact that after the fall of 1998, students were

not given a presentation of the computer instructional aid. Instructors simply informed the students of the availability of the aid and made copies if asked. This required the students to take the initiative and make the effort to obtain a copy of the aid from the instructor. Over two-thirds of the students used the aid outside of class during the fall of 1998. According to instructor's estimates, less than one-fifth requested a copy of the aid in later quarters. The effort seemed greater to students and the sensibility was less because of the lack of a presentation.

As the potential inside developer examines these concepts and considers a possible project, he or she needs to consider the need, sensibility and effort requirements as perceived by the instructors and the students. This will help the potential inside developer decide what his or her own perceptions of need, sensibility and effort are and whether to use or do this type of project.

Ideal Model

Assuming the inside developer already had a thorough knowledge of the organization, the problem, and its environment, the inside developer also needs substantial knowledge of instructional design, technology, especially educational technology, diffusion of innovation. The following checklist is based on what I learned in this case. The first part of the checklist, I through VI, is provided as a guideline to help a potential inside developer know what knowledge and skills will be needed at each stage of the development process. The potential inside developer can use the checklist to identify areas where additional knowledge and skills might be needed and if an inside developer might be the best under specific circumstances. The second part of the

checklist, VII and VIII, relates directly to features of the potential project which are important in deciding if the inside developer should undertake the project.

Checklist

I. CONDUCT PRELIMINARY DEVELOPMENT – Assess skills and status

A. Consider Organization and Environment

1. Find out decision-making processes

a. Do I know who makes decisions?

b. Do I understand how are decisions made?

2. Consider objectives of organization

a. Do I have the stated objectives of the organization?

b. Am I familiar with the unstated objectives of the organization?

c. Do I understand effect these objectives have on decision-making?

B. Consider Instructional Design

1. Do I know at least one modern model of instructional design?
2. Can I recognize weakness in design according to this theory?
3. Can I identify essential instructional elements?
4. Can I assess the consistency of instructional elements?

C. Consider Computer Technology

1. Have I had any programming experience?
2. Am I familiar with computer hardware?
3. Am I comfortable with basic operating systems?
4. Have I had a fair amount of experience with applications software?
5. Do I know what applications are suitable for instructional design functions?

D. Consider Diffusion of Innovation

1. Do I understand the process of diffusion of innovation?
2. Do I know the role of the change agent?
3. Can I "sell" an idea to colleagues?

II. PERFORM ANALYSIS

A. Analyze Desired Course Outcomes

1. Do I know the desired outcomes for the course?
2. Are there any unstated course outcomes? Do I know them?
3. Are those outcomes clear to me? Do I understand them?
4. Do I know the conditions, behaviors, criteria and lower limits of the course and unit requirements?

B. Analyze Course Content

1. Do I know what the content of the course is?
2. Do I teach the course content?
3. Do I know the typical models for organizing that course content?

C. Analyze Course

1. Have I taught this course several times?
2. Am I familiar with problems instructors have with this course?
3. Am I familiar with problems students have with this course?
4. Is this a standard multi-section course?

D. Analyze Instructors
    1. Do I have a list of teachers who have taught this course?
    2. Do I have a list of teachers who are currently teaching this course?
    3. Do I know what else is expected of those instructors?
    4. How many of these instructors do I know?
    5. Do I have a comfortable relationship with these instructors?

E. Analyze Students
    1. Am I familiar with the typical student who takes this course?
    2. Am I familiar with the atypical student who takes this course?
    3. Do I know what is covered in any prerequisites for this course?
    4. Do I know how students respond to various aspects of the course?

F. Analyze Your Personal Time Restrictions
    1. Analyze time for reflection
        a. Do I have time for reflection?
        b. Can I tradeoff this time with some lower priority task?
        c. Am I good at finding possible solutions to problems?
    2 Analyze time for communication with colleagues
        a. Do I have time for communicating with colleagues?
        b. Where is this time coming from?
        c. Am I comfortable talking with my colleagues?
        d. Are my colleagues comfortable talking with me?
        e. Do my colleagues have time to talk with me?

G. Analyze Physical Limitations of Environment (especially availability rooms and computers)
    1. Do I know what facilities are available?
    2. Do I know when those facilities are available?
    3. Do I know how obtain and reserve those facilities?

H. Analyze Existing Measurement Tools
    1. Do I have copies of any standard or "common" exams?
    2. Do I have copies of any typical "formative" quizzes?
    3. Do I have descriptions of typical projects and assignments?

I. Analyze Personal Objectivity Regarding the Above
    1. How closely do I associate myself with this course?
    2. Was I involved in creating this course or any of its components?
    3. How well do I accept suggestions and criticisms?

III. DESIGN SOLUTION TO INSTRUCTIONAL PROBLEM

A. Consider Knowledge of Organization - including knowledgeable and influential users
    1. What departments, branches, etc. of this organization have a connection to his course?
    2. Who are the gatekeepers for this course and its materials?
    3. Which instructors are the most knowledgeable about this course?
    4. Which instructors are the most influential regarding this course?

B. Consider Skill and Experience in Design

1. Am I experienced in using at least one system design model?
2. Am I familiar with the strengths and limitations of that model?
3. What instructional theory-based solution will solve the problem noted?
4. Will the solution be consistent with the rest of the system?
5. Will the solution appear useful to teachers and students?
6. Will the solution appear to be something needed by teachers and students?
7. Will the solution likely be successful based on your experience, theory and research?
8. Will the solution involve no more than the time and resources available?
9. Will the solution best involve computer application?
10. Can the product be self standing and require little or no effort on the part of the instructors?

IV. DEVELOP THE INSTRUCTIONAL SOLUTION

A. Consider Your Technology Skills

1. Check your familiarity with computers
    a. Am I familiar with different types of computer hardware needed for the solution?
    b. Am I familiar with common types of operating systems needed for the solution?
    c. Am I familiar with typical applications software needed for the solution?
    d. Am I familiar with the types of educational software needed for the solution?
2. Consider your programming skills
    a. Do I know at least one programming language suitable for this solution?
    b. Have I had experience creating a computer program from scratch?
    c. Have I had experience debugging computer programs?
    d. Am I familiar with common programming tools?

B. Consider the Time Available

1. Check time for development
    a. How much time will I have for development?
    b. Where is this time coming from?
    c. Will this amount of time change over the next few months?
    d. Can I find a way to get additional time, if needed?
2. Check time for testing and debugging
    a. How much time do I have for testing and debugging?
    b. Where is this time coming from?
    c. Will this amount of time change over the next few months?
    d. Can I find a way to get additional time, if needed?

V. IMPLEMENT SOLUTION

A. Consider Knowledge of Organizational Policies Affecting Changes

1. Do I know official policies concerning implementation of change?
2. Am I familiar with any recent successful changes?
3. Am I familiar with any recent changes that have failed?

B. Consider Knowledge of Experienced and Influential Users

1. Do I have a list of teachers who are currently teaching this course?
2. How many of these instructors do I know?
3. Which instructors are the most influential regarding this course?
4. Which instructors could sabotage any innovations or changes?

C. Assess Time Available for "Marketing" and Diffusion

1. How much time will I have for marketing and diffusion?
2. Where is this time coming from?
3. Will this amount of time change over the next few months?
4. Can I find a way to get additional time, if needed?

VI. EVALUATE & MAINTAIN SOLUTION

A. Consider Organization's Policies and Processes Regarding Change

1. Am I familiar with policies for evaluation of change?
2. Am I familiar with policies for maintaining change?
3. Are there any unstated policies regarding these?

B. Consider Assessment Procedures

1. Am I familiar with any assessment tools that are currently used?
2. Do I know how the results of these tools are evaluated?
3. Have I any measures to assess the perceived usefulness of the solution?
4. Can I create simple response forms or unobtrusive measures to assess the perceived usefulness of the solution?
5. Is the solution likely to be powerful enough to show a difference in learning results?
6. Who is likely to allow testing the solution in his or her class?
7. Will I be able to accommodate the timing of the class to apply the solution and test it?
8. Do I have a plan with permissions for gathering complete data from as many students as possible?
9. Do I have time and skill to summarize and report the results?

VII. ASSESS THE SIZE OF THE PROJECT

A. Consider the Portion of the Curriculum or Course Involved in the Project

1. Do I know the scope and boundaries of this project?

B. Consider the Overall Estimated Size of the Project

2. Do I have an overall estimate of time needed for this project?
3. Can I estimate how many lines of programming code are needed?

C. Consider Number of Potential Users of the Project

1. Do I know how many sections of this course there are?
2. Do I know how often this course is offered?
3. Do I know the average number of students in each section?

D. Consider Estimated Lifetime of the Product

1. Can I estimate how long this course will be continued?
2. Do I know how often major changes are made to this course?
3. Do I know of any possible changes that might affect the usefulness of this product?
4. Can I estimate the rate of turnover of instructors that are sold the product?

VII. ASSESS TIME REQUIRED

A. For Regular Duties and Responsibilities of your Position

1. Do I have an accurate estimate of the time my position requires?
2. How much do these time requirements vary?

B. For Personal Life

3. Can I decrease the amount of time in my personal life?
4. Do I want to decrease the amount of time in my personal life?

C. For this Project

1. Analysis
    a. Do I have a good estimate of the time expected for analysis?
    b. Do I know the time required for a worse case analysis scenario?
2. Design
    a. Do I have a good estimate of the time expected for design?
    b. Do I know the time required for a worse case design scenario?
3. Development
    a. Do I have a good estimate of the time expected for development?
    b. Do I know the time required for a worse case development scenario?
4. Implementation (and diffusion)
    a. Do I have a good estimate of the time expected for implementation?
    b. Do I know the time required for a worse case implementation scenario?
5. Evaluation and maintenance
    a. Do I have a good estimate of the time expected for evaluation and maintenance?
    b. Do I know the time required for a worse case evaluation and maintenance scenario?
6. Overall
    a. Can I accept the fact that the above worse case scenarios are probably going to happen?

Advantages and Disadvantages of Insider and Outsider

In evaluating the results of this study I reexamined Havelock's list (1973) of the advantages and disadvantages of an insider versus an outsider (Table 2.5). Two of Havelock's three insider disadvantages and matching outsider disadvantages were not confirmed by this study. These are indicated with a D, for denied, rather than a C, for confirmed, in the first column of Table 5.1. The first was that the insider "may have to live down his past failures" (p. 51) while the outsider "starts fresh" (p. 52). Personally, I did not detect any problems in this area. This may have been because I was not sensitive to this problem, no one confronted me directly about past failures or because I had no such problem. This is not to say that this is not a potential problem for an inside developer, simply that this study did not confirm this disadvantage.

The second insider disadvantage denied, or at least not confirmed, by this study related to Havelock's statement that "the inside change agent usually faces the difficult task of redefining his on-going relationships with the other members of the system" (p. 51). I knew two of the instructors in this study quite well, Lori and Joyce. If anything, this project seemed to benefit from my prior relationships with these instructors. Of the other instructors, Judy was a passing acquaintance and I had no prior relationship with Tina or Cynthia. All three seemed to be quite willing to let me start fresh. Some of this attitude may have been due to the fact that, since I was a colleague, I was "independent of the power structure" (p. 52) much as an outsider would have been. In fact, since administrators one or more levels above the user probably hired the outsider, the outsider's relationship with the users might suffer from identification with the power structure.

Table 5.1
Confirmation and Denial of Features of Inside and Outside Change Agents

	INSIDER ADVANTAGES	OUTSIDER DISADVANTAGES
C	Knows system	The outsider may lack the knowledge of the insider
C	Speaks the language	
C	Understands the norms	
C	Identifies with the system's needs and aspirations	May not "care enough"
C	Is a familiar figure	Is a stranger
	INSIDER DISADVANTAGES	**OUTSIDER ADVANTAGES**
C	May lack perspective	Is in a position to have perspective
C	May not have the special knowledge or skill	Is in a position to bring in something genuinely new
D	May have to live down his past failures	Starts fresh
C	May not have an adequate power base	Is independent of the power structure
C	May not have the independences of movement	
D	The inside change agent usually faces the difficult task of redefining his on-going relationships with the other members of the system	

In addition to the factors Havelock identified, I identified several additional factors as a result of this study. These are identified in Table 5.2 with an A in the first column. Havelock did not specifically list the insider's disadvantage of limited time. However, under the insider's disadvantage of limited independence of movement, he did state that "the obligations of membership may severely limit the time and energy that he can invest in his new role" (p. 51).

Table 5.2
Additional Features of Inside and Outside Change Agents

	INSIDER ADVANTAGES	OUTSIDER DISADVANTAGES
A	Does not need to spend much time studying the problem and situation	Must spend a substantial amount of time analyzing the problem and situation
A	Available for long-term maintenance and upgrading	Probably will not be available for long-term support
A	Fully aware of potential problems involving diffusion	Does not know the people or organizational culture that can cause problems with diffusion
A	Can insure a component is consistent with rest of system	May have difficulty insuring a component is consistent
	INSIDER DISADVANTAGES	**OUTSIDER ADVANTAGES**
A	May have limited time to devote to project	Usually can work full-time on project
A	Limited to relatively small projects	Has access to additional time and assistance needed for large projects

Further Research

Prior to the development of a computer instructional aid, it is important that the developer have a good understanding of (1) the organization and environment; (2) instructional design; (3) computer technology, especially in education; and (4) the diffusion of innovation. To substantiate this claim, several questions still need to be answered.

- To what extent can understanding of the above four areas be taught?
- What is the best mix between knowledge of the organization, the insider's advantage, and knowledge of the system development processes of instructional design, technology and change, the outsider's advantage?

- Can an insider be more accurate at identifying problems than an outsider?
- Is it possible to provide aids that would substitute for lengthy training?
- Could a college provide in-service training that would be productive in providing more inside developers?
- What are the advantages and disadvantages of different technologies?
- How does one choose a specific technology to solve a particular problem?
- Is it possible to provide aids to help make these decisions regarding appropriate technologies
- As new technologies are introduced and old technologies are improved, when and how do we decide if they can be used?

During the stage of analysis, reflection and communication are important.

- How much time do instructors spend considering and discussing their experiences and problems?
- Could there be a relationship between this amount of time and the tendency to initiate changes and innovations?
- Will a check with colleagues provide greater objectivity to an insider?
- What are the differences in designing a small component of a system and developing an entire system?
- How much time is required in the design stage?
- Is the amount of analysis time related to the experience of the developer?
- Under what conditions can an insider's analysis be as useful as that of a team?

Many of the questions related to design could also be related to development. For example:

- What size project is a sole insider more efficient than a dedicated outsider, perhaps working with a team?
- How can the insider reliably estimate how much time a project will take?
- Is the amount of development time related to the experience of the developer?
- What variables influence the amount of development time?
- Under what conditions can an inside developer be as successful as a team?

There are also several areas of implementation that can justify additional research.

- To what extent is the insider's knowledge and experience of the organization and its people a benefit?
- What are the insider's disadvantages during implementation?
- How much time needs to be allocated for implementation?

Additional research in the area of evaluation and maintenance might include questions such as:

- How much maintenance of a computer instructional aid is required?
- What factors affect the amount of maintenance of a computer instructional aid?
- What is the lifetime of a computer instructional aid? What determines lifetime?

Researchers need to look more closely at the differences between inside developers and outside developers.

- What are the specific differences between developing a complete course and only one component of a course?
- Finally, how much knowledge is required in the areas of instructional design, technology, educational technology and diffusion of innovation, to be a successful inside developer?

Final Conclusions

An inside developer creating a computer instructional aid is not a trivial pursuit. It requires a great deal of knowledge, time and effort. The inside developer needs to know quite a bit about instructional design, education and technology, diffusion of innovation, and the organization and content involved. Can an inside developer be successful at creating a computer instructional aid? Yes, when there is adequate need of a sensible product to justify the effort involved. Yes, when the application is worth the effort, such as when the product can be of use to many instructors. Yes, when the insider developer already knows the organization and content involved. Yes, when the inside developer has enough background and experience in instructional design to troubleshoot weaknesses in a course and to judge the system consistency of newly developed parts; enough knowledge of technology to determine when to apply computer technology and to do the programming necessary for the aid; and enough ideas about diffusion to create a product acceptable and useful to colleagues and students. This should help others understand the amount of background, experience, time and effort that are required. Each situation and

potential inside developer is unique. However, a potential inside developer should now have a better idea of what must be done before a similar project is undertaken, what types of projects are suitable, and how to proceed through the design, development and diffusion of that project.

APPENDICES

APPENDIX A

Early Pilot-Type Program (1988)

```
5 REM PILOT IN BASIC     JIM NEWTOWN    1/18/89
10 REM
15 REM BASIC PROGRAM TO RUN SIMPLE PILOT PROGRAM
20 REM
25 REM VARIABLES:
30 REM IN$       INPUT LINE FROM ASCII FILE
35 REM INFILE$   INPUT ASCII TEXT FILE
40 REM ANS$      ANSWER ENTERED BY STUDENT
45 REM MATCH$    MATCH FROM INPUT FILE
50 REM FLAG      1 IS A MATCH, 0 IS NOT MATCH
55 REM LABEL$    LABEL FROM MODULE OR SECTION
60 REM PST       POSITION IN STRING
100 REM
110 GOSUB 1000                                    'INITIALIZATION MODULE
120 IF LEFT$(IN$,2)="A:" THEN GOSUB 2000:GOTO 180 'ANSWER MODULE
130 IF LEFT$(IN$,2)="M:" THEN GOSUB 3000:GOTO 180 'MATCHING MODULE
140 IF LEFT$(IN$,2)="J:" OR LEFT$(IN$,3)="JY:" OR LEFT$(IN$,3)="JN:"
THEN GOSUB 4000                                          'JUMP MODULE
150 IF LEFT$(IN$,2)="E:" THEN 9050
160 IF LEFT$(IN$,2)="R:" THEN 180                 'REMARK, JUST CONTINUE
170 GOSUB 6000                                    'PRINT MODULE
180 GOSUB 8000                                    'READ INPUT FILE MODULE
190 GOTO 120
1000 REM INITIALIZATION MODULE
1010 CLS
1020 PRINT "BASIC PROGRAM TO RUN A SIMPLE PILOT PROGRAM IN ASCII FORMAT"
1030 PRINT "PLEASE ENTER PILOT FILE NAME INCLUDING DISK DRIVE
DESIGNATION"
1040 INPUT INFILE$
1050 OPEN INFILE$ FOR INPUT AS #1
1060 GOSUB 8000
1070 LET CTR=0
1080 RETURN
2000 REM ACCEPT ANSWER MODULE
2010 INPUT ANS$
2020 RETURN
3000 REM MATCH ANSWER MODULE
3010 REM DETERMINE MATCH$
3020 LET PST=3
3030 LET MATCH$=""
3040 LET Z=0
3050 IF PST>LEN(IN$) THEN 3150
3060 IF MID$(IN$,PST,1)=" " THEN 3100
3070 LET MATCH$=MATCH$+MID$(IN$,PST,1)
3080 LET Z=1
3090 GOTO 3110
3100 IF Z=1 THEN 3150
3110 LET PST=PST+1
3120 GOTO 3050
3150 REM CHECK FOR MATCH
3160 IF MATCH$=ANS$ THEN FLAG=1 ELSE FLAG=0
3170 RETURN
```

```
4000 REM JUMP MODULE
4010 IF FLAG=1 AND LEFT$(IN$,3)="JY:" THEN 4100
4020 IF FLAG=0 AND LEFT$(IN$,3)="JN:" THEN 4100
4030 IF LEFT$(IN$,2)="J:" THEN 4100
4040 GOTO 4300
4100 REM GET LABEL NAME
4110 LET PST=3
4120 LET LABEL$=""
4130 LET Z=0
4150 IF MID$(IN$,PST,1)=":" THEN 4200
4160 IF MID$(IN$,PST,1)=" " THEN 4200
4170 LET LABEL$=LABEL$+MID$(IN$,PST,1)
4180 LET Z=1
4190 GOTO 4210
4200 IF Z=1 THEN 4300
4210 LET PST=PST+1
4220 GOTO 4150
4300 REM GET LABEL NAME
4310 GOSUB 8000
4320 IF MID$(IN$,1,LEN(LABEL$))=LABEL$ THEN 4400
4330 GOTO 4310
4400 RETURN
6000 REM PRINT MODULE
6010 IF MID$(IN$,1,3)="TY:" AND FLAG=0 THEN 6200
6020 IF MID$(IN$,1,3)="TN:" AND FLAG=1 THEN 6200
6023 IF LEFT$(IN$,3)="TY:" OR LEFT$(IN$,3)="TN:" THEN
IN$=RIGHT$(IN$,LEN(IN$)-3)
6025 IF LEFT$(IN$,1)="*" THEN 6200
6030 PRINT IN$
6040 LET CTR=CTR+1
6050 IF CTR<20 THEN 6200
6100 REM STOP SCROLL
6110 PRINT
6120 PRINT "PRESS ENTER TO CONTINUE"
6130 INPUT Z$
6140 LET CTR=0
6200 RETURN
8000 REM READ INPUT FROM FILE
8010 IF EOF(1) THEN 9000
8020 INPUT #1, IN$
8030 RETURN
9000 REM END OF FILE REACHED
9010 CLOSE #1
9020 OPEN INFILE$ FOR INPUT AS #1
9030 INPUT #1, IN$
9040 RETURN
9050 REM END OF PROGRAM MODULE
9055 PRINT "END OF PROGRAM"
9060 CLOSE #1
9999 END
```

APPENDIX B

Early Text File (1989)

```
01. BASIC command that executes the program.
     A.   RUN                      C.   LOAD
     B.   SAVE                     D.   LIST
A:
M:A
TY:CORRECT
TN:INCORRECT
02. The names assigned to memory locations in primary storage.
     A.   Condition                C.   Variables
     B.   Line numbers
A:
M:C
TY:CORRECT
TN:INCORRECT
03. BASIC statement that instructs the computer to accept values and
    assign them to variables.
     A.   LET                      D.   INPUT
     B.   END                      E.   REM
     C.   PRINT
A:
M:D
TY:CORRECT
TN:INCORRECT
04. BASIC statements used in structured programming to transfer control
    from one module to another, and back to the first when the second
    is completed.
     A.   IF/THEN/ELSE             C.   GOSUB/RETURN
     B.   WHILE/WEND
A:
M:C
TY:CORRECT
TN:INCORRECT
05. BASIC command that displays program lines on the monitor.
     A.   SAVE                     C.   LOAD
     B.   RUN                      D.   LIST
A:
M:D
TY:CORRECT
TN:INCORRECT
06. BASIC statement used to assign values and the results of
    calculations to variables.
     A.   END                      D.   REM
     B.   INPUT                    E.   PRINT
     C.   LET
A:
M:C
TY:CORRECT
TN:INCORRECT
07. BASIC statement used to produce output.
     A.   LET                      D.   INPUT
     B.   END                      E.   REM
     C.   PRINT
```

A:
M:C
TY:CORRECT
TN:INCORRECT
08. BASIC command that copies the program from primary storage to auxiliary storage.
A. LIST C. RUN
B. LOAD D. SAVE
A:
M:D
TY:CORRECT
TN:INCORRECT
09. BASIC statement used for internal documentation, especially at the beginning of modules.
A. INPUT D. REM
B. END E. LET
C. PRINT
A:
M:D
TY:CORRECT
TN:INCORRECT
10. BASIC command that copies the program from auxiliary storage to primary storage.
A. SAVE C. RUN
B. LIST D. LOAD
A:
M:D
TY:CORRECT
TN:INCORRECT
11. BASIC statement that uses the status of a condition to select which of two sets of instructions are to be performed.
A. GOSUB/RETURN C. IF/THEN/ELSE
B. WHILE/WEND
A:
M:C
TY:CORRECT
TN:INCORRECT
12. BASIC statements that use the status of a condition to determine whether looping should continue or end.
A. GOSUB/RETURN C. WHILE/WEND
B. IF/THEN/ELSE
A:
M:C
TY:CORRECT
TN:INCORRECT
13. An expression in an IF/THEN or WHILE/WEND statement that the computer evaluates. If true, the computer follows one set of instructions, if false, another set of instructions.
A. Condition C. Variables
B. Line numbers
A:
M:A
TY:CORRECT
TN:INCORRECT
E:

APPENDIX C

Adult Learning Lesson (1990)

```
*PAGE1
            Welcome to the Budgeting Tutorial.
     This program was written by Jim Newtown using information in the
bulletin "Budget Basics" from the Michigan State University Cooperative
Extension Service.
     This program will be run a little differently than many tutorials
presented with Programmed Instruction.  To begin with, you will first be
asked questions.  Choose one of the answers by entering the letter in
front of it.  Don't worry about answering them correctly.  Scores will
not be kept.
NOTE:  PLEASE PRESS CAPS LOCK BUTTON UNTIL LIGHT COMES ON.

*PAGE2
What type of budget system should you create?
     A    A complete one
     B    A complex one
     C    A simple one
     D    A general one
     E    One that someone else takes care of
A:
M:C
JN:*PAGE3
JY:*PAGE4

*PAGE3
No, you should create a simple system.
J:*PAGE5

*PAGE4
Good, a simple system is the best system.
J:*PAGE5

*PAGE5
     Decide who will be responsible for paying bills and keeping
records.  One partner may take this responsibility, or you may decide to
share it.  Whatever you decide, be sure you both understand who is
responsible for what.

*PAGE6
What is most important regarding keeping records?
     A    Keep complete records
     B    Keep detailed records
     C    Keep accrual basis records
     D    Keep accurate records
     E    Keep all pop records but discard rock and roll records
A:
M:D
JN:*PAGE7
JY:*PAGE8

*PAGE7
No, while that is important, keeping accurate records is most
```

```
important.
J:*PAGE9

*PAGE8
Yes, keeping accurate records is most important.
J:*PAGE9

*PAGE9
     To keep track of expenses, retain all bill stubs and receipts.
Write on the back of the receipt the category of the expense, such as
food, household supplies or entertainment.  This is particularly
important for flexible expenses because you will need to look at your
checkbook to estimate current spending.  Keep receipts in an envelope
and record them weekly in each category.  Set aside a time to
record your weekly expenses.

*PAGE10
How should expenses be organized?
     A    By type (fixed, flexible and periodic)
     B    By amount (largest first)
     C    By name (alphabetically with A's first)
     D    By date (chronologically with oldest first)
     E    By who you borrowed money from to pay them.
A:
M:A
JN:*PAGE11
JY:*PAGE12

*PAGE11
No, while that might be helpful, prioritizing them by type is more
important.
J:*PAGE13

*PAGE12
Yes, prioritizing expenses by type is the most important.
J:*PAGE13

*PAGE13
     Because your income is fixed, you must keep your expenses in line.
First subtract your fixed and fixed periodic expenses from your net
income.  The amount left will  be for flexible and flexible periodic
expense.
     At the end of the month record the actual amounts you have spent.
If you are consistently over-or under spending in a category, reevaluate
your budget.  It is better to overestimate expenses -- this will give
you a cushion to work with.

*PAGE14
When should you review your budget?
     A    Randomly
     B    Periodically
     C    Seldom
     D    Weekly
     E    You're supposed to review your budget?
A:
M:B
TY:Yes, that's right, you should review your budget randomly.
```

```
M:A
TY:No, budgets should be reviewed periodically, not randomly.
M:C
TY:No, budgets should be reviewed periodically, usually on a
TY:monthly basis.
M:D
TY:No, weekly is probably too often.  Usually every month or
TY:so is sufficient.  It's more important just to review your
TY:budget preiodically.
M:E
TY:You're in real trouble.  You should see one of our
TY:counselors immediately.
JY:*PAGE18

*PAGE15
     Once you start recording expenses you may be surprised at how much
you spend and where you spend it.  This system will allow you to
evaluate your spending habits and determine if you need to make changes
in these habits.
     After several months, spending patterns will emerge. Note each
month where expenses were out of the ordinary.  You may find that each
month there is something that wasn't planned.  To deal with this, you
will want to make your budget flexible.  Remember, your income doesn't
usually change but your expenses can.

*PAGE16
The most important part of budgeting is:
     A    Keeping accurate records
     B    Being systematic
     C    Communicating
     D    Concentrating on self-discipline
     E    Knowing when to skip town
A:
M:C
TN:No, that might help but...
TY:Yes, that's right...
     The most important part of budgeting is communicating. Openly
discuss your situation.  If you have questions, ask them.  This will
help you stay organized and avoid problems or misunderstandings.

*PAGE17
IN CONCLUSION:  GIVE YOUR BUDGET A CHANCE TO WORK.
     Remember to keep your system simple but organized.  You can only
benefit from a workable spending plan.

*PAGE18
Would you like to:
     A    Start this program again from the beginning
     B    End this program
A:
M:A
JY:*PAGE1
M:B
TY:Good Bye.
*END
```

APPENDIX D

Course Outcomes

INF111: INFORMATION SYSTEMS THEORY

2 Quarter Hours

Course Outcomes

The student will demonstrate knowledge of the following subject matter, as measured on instructor tests and quizzes with a minimum of 60% accuracy:

1. Terminology and principles related to hardware used in computer information systems including:
   a. Types of input hardware.
   b. Processor unit components including CPU and main memory.
   c. Types of output hardware.
   d. Types of auxiliary storage.
2. Terminology and principles related to software used in computer information systems including:
   a. System programs.
   b. Application programs.
   c. Operating environments and user interfaces.
3. Terminology and principles related to data and data storage in computer information systems including:
   a. Data to information processing cycle.
   b. Data hierarchy and organization.
   c. Batch and interactive processing.
   d. Coding and number systems (ASCII, binary, etc.)
   e. Measurement of storage space (bit, byte, etc.)
   f. Features of data storage including volatile and nonvolatile, memory addresses, and RAM vs. ROM.

4. Terminology and principles related to the development of a computer information system including:

   a. Process of developing a system (e.g. System Development Cycle and "How to buy a computer.")
   b. Users and computer specialists involved in the creation and operation of computer information systems.

   c. Acquisition of hardware including terms and measurements used in specifications (such as those found in a typical advertisement).

   d. Acquisition of software including the computer programming process and terms used in software requirements.

   e. Common methods and problems involved in conversion and implementation of new systems.

5. Terminology and principles related to other computer information systems topics including:
   a. Data communications and networks.

   b. History, generations and trends in the development of information processing systems.
   c. Important issues such as ergonomics, security, productivity and integration of information processing systems.

Critical Literacy skills are addressed in CO's #4a-e.

Information Literacy skills are not addressed in these CO's.

Cultural Diversity awareness is not addressed in these CO's.

Global Awareness is not addressed in these CO's.

APPENDIX E

Program Including Portion of Test Bank

```
100 REM INF111.BAS   JIM NEWTOWN 9/13/98
110 REM CAI PROGRAM
120 REM VARIABLES:
130 REM    A  =NUMBER OF QUESTIONS IN SELECTED CHAPTER
140 REM    D  =NUMBER OF DATA LINES IN SELECTED CHAPTER
150 REM    NL =RANDOM NUMBER LIST
160 REM        COL. 1 = RANDOMIZED QUESTION NUMBER (1 TO A)
170 REM        COL. 2 = 1 MEANS ANSWERED CORRECTLY, 0 MEANS NOT
180 REM    Q$ =ARRAY HOLDING DATA LINES FOR SELECTED CHAPTER
190 REM    G$ =STUDENT GUESS
200 REM    A$ =CORRECT ANSWER
210 REM *** INITIALIZATION ******************
220 KEY OFF
230 CLS
240 DIM Q$(500): DIM NL(2,100)
250 INPUT "What is your name";N$
260 REM *** READ MENU AND PRINT MENU ********
270 LET CHP=0
280 GOSUB 2010
290 REM *** PRINT MENU ***********************
300 IF CHP<>0 THEN 430
310 CLS
320 PRINT RIGHT$(Q$(1),LEN(Q$(1))-4)
330 FOR X=2 TO (D-1)
340   PRINT Q$(X)
350 NEXT X
360 REM *** SELECT CHAPTER **********************
370 INPUT "Which chapter would you like to review"; CHP
380 IF CHP=0 THEN 370
390 IF CHP=90 GOTO 2220
400 IF CHP=99 THEN 1070
410 REM *** READ CHAPTER QUESTIONS **************
420 GOSUB 2010
430 REM *** SET UP RANDOM LIST ***********************
440 PRINT: PRINT
450 RANDOMIZE TIMER
460 FOR Y=1 TO A
470    LET Z=INT(RND*(A))+1
480    LET MATCH$="YES"
490    FOR X = 1 TO Y
500       IF NL(1,X) = Z THEN MATCH$="NO"
510    NEXT X
520    IF MATCH$<>"NO" THEN LET NL(1,X-1)=Z ELSE 470
530 NEXT Y
540 LET TRYS=0: LET NC=0
550 LET T1=VAL(MID$(TIME$,1,2))*60 + VAL(MID$(TIME$,4,2))
560 REM *** QUESTION LOOP - FIND QUESTION ********
570 LET FIN$="YES"           :REM INDICATOR FOR ALL QUESTIONS ANSWERED
580 LET Z=1                  :REM READ A NOT CORRECT QUE FROM LIST
590 IF NL(2,Z)=1 THEN 920    :REM THIS QUESTION ALREADY ANSWERED
600   LET FIN$="NO"          :REM IND. - NOT ALL QUESTIONS ANSWERED YET
610   LET X=0: Y=1
```

```
620   IF LEFT$(Q$(Y),1)="~" THEN X=X+1
630     IF X=NL(1,Z) THEN 670
640     LET Y=Y+1
650     IF Y>D THEN PRINT "PROGRAM ERROR - QUESTION NOT FOUND"
660   GOTO 620
670        PRINT RIGHT$(Q$(Y),LEN(Q$(Y))-4) :REM PRINT FIRST QUE LINE
680        LET Y=Y+1            :REM INCREMENT Y AND CONTINUE QUESTION
690   REM *** LOOP TO PRINT QUESTION ***
700   IF LEFT$(Q$(Y),1)="~" THEN 790
710     REM *** NOT FIRST LINE ******************
720     IF LEFT$(Q$(Y),1)<>"*" THEN 760
730       LET A$=MID$(Q$(Y),2,1)
740       PRINT RIGHT$(Q$(Y),LEN(Q$(Y))-1)
750     GOTO 770
760       PRINT Q$(Y)
770   LET Y=Y+1               :REM INCREMENT Y AND CONTINUE QUESTION
780   GOTO 690
790     REM *** FIRST LINE OF NEXT QUESTION ***
800     REM *** END OF QUESTION - GET ANSWER ***
810       LET FIRST$="YES"      :REM FIRST TRY FOR THIS QUESTION
820       LET TRYS=TRYS+1
830         INPUT "ENTER CHOICE (Q to quit)"; G$
840         IF G$="Q" THEN 960
850         IF A$=G$ THEN 890
860         IF G$=CHR$(ASC(A$)+32) THEN 890
870           PRINT "Incorrect, try again": LET FIRST$="NO": PRINT
880         GOTO 820               :REM TRY AGAIN
890           PRINT "Very good - correct":PRINT
900           IF FIRST$="YES" THEN LET NL(2,Z)=1 :REM CORRECT FIRST TRY
910           IF TRYS<=A THEN LET NC=NC+1
920           IF Z >=A THEN 950  :REM QUESTION DONE, GO NEXT QUESTION
930             LET Z=Z+1        :REM NEXT QUESTION
940             GOTO 590
950 IF FIN$="NO" THEN 560
960 REM *** CONCLUSION ************************************************
970 PRINT "CAI TEST - CHAPTER "; CHP, DATE$, TIME$
980 PRINT N$; ", you got "; NC; " correct on the first attempt."
990 IF G$<>"Q" THEN FIN$="YES" ELSE FIN$="NO": GOTO 1010
1000 PRINT "It took you "; TRYS; " trys to answer all "; A; " questions
     correctly."
1010 REM *** RECORD CHAPTER RESULTS ***********
1020 LET T2=VAL(MID$(TIME$,1,2))*60 + VAL(MID$(TIME$,4,2))
1030 LET T=(T2-21) * 7 * CHP
1040 OPEN "A:RESULTS" FOR APPEND AS #1
1050 WRITE #1, N$, CHP, FIN$, DATE$, NC, TRYS, T
1060 CLOSE #1
1070 REM *** END OF PROGRAM ******************
1080 PRINT: PRINT "Would you like to end the program?"
1090 INPUT "Y or N"; RES$
1100 IF RES$<>"Y" THEN 260
1110 SYSTEM
1120 END
2000 REM ***************************************************************
2010 REM *** SUBROUTINE: FIND CHAPTER IN DATA STATEMENTS ***
2020 RESTORE
2030 READ P$
2040 IF LEFT$(P$,1)<>"~" THEN 2030
```

```
2050 IF MID$(P$,2,3)<>"END" THEN 2080
2060     PRINT "CHAPTER NUMBER NOT FOUND, TRY AGAIN"
2070     GOTO 2210
2080 IF VAL(MID$(P$,2,1))<>CHP THEN 2030
2090 REM *** READ CHAPTER QUESTIONS INTO ARRAY Q$ ***
2100   LET Q$(1)=P$
2110   LET A=1
2120   LET D=2
2130 READ Q$(D)
2140   IF LEFT$(Q$(D),1)<>"~" THEN 2180
2150   IF MID$(Q$(D),2,3)="END" THEN 2200
2160   IF VAL(MID$(Q$(D),2,1))<>CHP THEN 2200
2170   LET A=A+1
2180   LET D=D+1
2190 GOTO 2130
2200 LET Q$(D)="~END"
2210 RETURN
2220 REM *** SUBROUTINE: PRINT RESULTS ***
2230 LPRINT ,"INF111 REVIEW PROGRAM RESULTS": LPRINT
2240 OPEN "A:RESULTS" FOR INPUT AS #2
2250 IF EOF(2) THEN 2330
2260 INPUT #2, NP$, CHPP, FINP$, DATP$, NCP, TRYSP, TP
2270 IF FINP$="NO" THEN 2300
2280   LPRINT NP$, CHPP, "COMPLETED", DATP$
2290 GOTO 2310
2300   LPRINT NP$, CHPP,,DATP$
2310 LPRINT ,NCP, TRYSP, TP
2320 GOTO 2250
2330 CLOSE #2
2340 GOTO 260
3000 REM ************************************************************
3001 DATA "~001These review questions cover the following chapters."
3002 DATA "    1. An Overview of Computer Concepts           (36
      questions)"
3003 DATA "    2. Computer Software Applications: User Tools (35
      questions)"
3004 DATA "    3. Input to the Computer                      (21
      questions)"
3005 DATA "    4. The System Unit                            (38
      questions)"
3006 DATA "    5. Output from the Computer                   (20
      questions)"
3007 DATA "    6. Secondary Storage                          ( 7
      questions)"
3008 DATA "    7. Communications and Networks                (13
      questions)"
3009 DATA " "
3010 DATA "   90. Print results"
3011 DATA " "
3012 DATA "   99. To quit"
3013 DATA " "
3014 DATA "~101An electronic device, operating under the "
3015 DATA "control of instructions stored in its own memory "
3016 DATA "unit, that can accept data (input), process data "
3017 DATA "arithmetically and logically, produce output from "
3018 DATA "the processing, and store the results for future "
3019 DATA "use is a(n)"
```

```
3020 DATA "*A. Computer"
3021 DATA "B. Calculator"
3022 DATA "C. Adding machine"
3023 DATA "D. Main memory"
3024 DATA ""
3025 DATA "~102Where are programs and data stored when they "
3026 DATA "are NOT being processed?"
3027 DATA "A. Main memory"
3028 DATA "B. Primary storage"
3029 DATA "C. RAM (random access memory)"
3030 DATA "*D. Auxiliary storage"
3031 DATA ""
3032 DATA "~103Computers that are more powerful than "
3033 DATA "microcomputers and can support a number of users "
3034 DATA "performing different tasks are called"
3035 DATA "A. Personal computers"
3036 DATA "B. Small computers"
3037 DATA "C. Supercomputers"
3038 DATA "*D. Minicomputers"
3039 DATA "   "
3040 DATA "~104The most powerful and expensive category of "
3041 DATA "computers that can process hundreds of millions "
3042 DATA "of instructions per second and perform long, "
3043 DATA "complex calculations are called"
3044 DATA "*A. Supercomputers"
3045 DATA "B. Mainframe computers"
3046 DATA "C. Minicomputers"
3047 DATA "D. Microcomputers"
3048 DATA ""
3049 DATA "~105The unit that contains the control unit and "
3050 DATA "the arithmetic logic unit is called the"
3051 DATA "A. Peripheral"
3052 DATA "B. CALU"
3053 DATA "*C. CPU"
3054 DATA "D. Calculator"
3055 DATA ""
3056 DATA "~106The major components of first generation "
3057 DATA "computers were"
3058 DATA "A. Transistors"
3059 DATA "B. Integrated circuits"
3060 DATA "*C. Vacuum tubes"
3061 DATA "D. Microprocessors"
3062 DATA "   "
3063 DATA "~107A computer hardware component used to enter "
3064 DATA "data into a computer is a(n)"
3065 DATA "A. Processor unit"
3066 DATA "B. Output device"
3067 DATA "*C. Input device  "
3068 DATA "D. Auxiliary storage unit"
3069 DATA ""
3070 DATA "~108A computer hardware component used to display "
3071 DATA "information is a(n)"
3072 DATA "*A. Output device"
3073 DATA "B. Input device"
3074 DATA "C. Processor unit"
3075 DATA "D. Auxiliary storage unit"
3076 DATA ""
```

```
3077 DATA "~109The major components of second generation "
3078 DATA "computers were"
3079 DATA "A. Integrated circuits"
3080 DATA "*B. Transistors"
3081 DATA "C. Vacuum tubes"
3082 DATA "D. Microprocessors"
3083 DATA ""
3084 DATA "~110The computer operations of input, process, "
3085 DATA "output and storage are collectively called:"
3086 DATA "*A. The information processing cycle"
3087 DATA "B. Data"
3088 DATA "C. The processor unit"
3089 DATA "D. A computer"
3090 DATA ""
3091 DATA "~111The part of the computer system that "
3092 DATA "electronically stores data and program "
3093 DATA "instructions when they are being processed is "
3094 DATA "called"
3095 DATA "*A. Main memory"
3096 DATA "B. Auxiliary storage"
3097 DATA "C. Control unit"
3098 DATA "D. ALU"
3099 DATA ""
3100 DATA "~112Input devices, output devices, and auxiliary "
3101 DATA "storage units that surround the processing unit "
3102 DATA "are sometimes referred to as"
3103 DATA "A. Electronic devices"
3104 DATA "B. Computers"
3105 DATA "C. Control units"
3106 DATA "*D. Peripherals"
3107 DATA ""
3108 DATA "~113In the _____ phase, the computer "
3109 DATA "manipulates data to create information."
3110 DATA "A. Input"
3111 DATA "*B. Processing"
3112 DATA "C. Output"
3113 DATA "D. Storage"
3114 DATA "   "
3115 DATA "~114What is another common term for a personal "
3116 DATA "computer?"
3117 DATA "A. Mainframe"
3118 DATA "B. Supercomputer"
3119 DATA "C. Minicomputer"
3120 DATA "*D. Microcomputer"
3121 DATA ""
3122 DATA "~115The device that contains the electronic "
3123 DATA "circuits that cause the processing of data to "
3124 DATA "occur, and includes the central processing unit "
3125 DATA "(CPU), main memory, and other electronic "
3126 DATA "components is called"
3127 DATA "A. Input device"
3128 DATA "B. Output device"
3129 DATA "*C. System unit"
3130 DATA "D. Auxiliary storage unit"
3131 DATA ""
3132 DATA "~116A detailed set of instructions that tells the "
3133 DATA "computer exactly what to do is called"
```

```
3134 DATA "A. Hardware "
3135 DATA "*B. Program"
3136 DATA "C. Data"
3137 DATA "D. Procedures"
3138 DATA "   "
3139 DATA "~117A complete electronic circuit that has been "
3140 DATA "etched on a small chip of nonconducting material "
3141 DATA "such as silicon is called a (n)__________?"
3142 DATA "A. Vacuum tube"
3143 DATA "B. Transistor"
3144 DATA "*C. Integrated circuit"
3145 DATA "D. Motherboard"
3146 DATA ""
3147 DATA "~118The unit that performs math and logic "
3148 DATA "operations is called the __________."
3149 DATA "*A. ALU"
3150 DATA "B. Primary storage"
3151 DATA "C. Auxiliary storage"
3152 DATA "D. Control unit"
3153 DATA ""
3154 DATA "~119Persons who design, write, test, and "
3155 DATA "implement programs necessary to direct the "
3156 DATA "computer to process data into information are "
3157 DATA "referred to as computer __________."
3158 DATA "A. Users"
3159 DATA "*B. Programmers"
3160 DATA "C. Operators"
3161 DATA "D. System analysts"
3162 DATA ""
3163 DATA "~120Most reports about computer errors are "
3164 DATA "usually traced to __________."
3165 DATA "A. Equipment failure"
3166 DATA "B. Software problems"
3167 DATA "C. Storage limitations"
3168 DATA "*D. Human mistakes"
3169 DATA ""
3170 DATA "~151The central processing unit (CPU) contains a "
3171 DATA "control unit that _________."
3172 DATA "A. Temporarily stores data and program "
3173 DATA "instructions they are being processed"
3174 DATA "*B. Executes the program instructions"
3175 DATA "C. Stores instructions and data when they are not "
3176 DATA "being used by the system unit"
3177 DATA "D. Performs math and logic operations"
3178 DATA ""
3179 DATA "~152The category of computers including hand-"
3180 DATA "held, notebook, laptop, portable, and desktop "
3181 DATA "computers is"
3182 DATA "A. Minicomputers"
3183 DATA "*B. Microcomputers"
3184 DATA "C. Mainframe computers"
3185 DATA "D. Small computers"
3186 DATA ""
3187 DATA "~153The production of information by processing "
3188 DATA "data on a computer is called:"
3189 DATA "A. A computer program"
3190 DATA "*B. Information processing"
```

```
3191 DATA "C. Input"
3192 DATA "D. Storage"
3193 DATA ""
…
4407 DATA "~END"
```

APPENDIX F

Student Instructions – Version 1

INF111 INFORMATION SYSTEMS THEORY
Instructions for Review Program

TO RUN PROGRAM FROM NETWORK (ROOM 105)

Click on **Start** in Windows 95
Click on **Programs** in Start menu
Click on **MS-DOS Prompt** in Programs menu
Type **F:** and then press the <Enter> button
Type **F:\ASSIGN\CAI** and then press the <Enter> button
Type **GWBASIC INF111** and then press the <Enter> button

If you receive a message that says **Device Fault in 940** then
Type **RUN** and press the <Enter> button to restart program

After exiting the program,
Type **Exit** and press the <Enter> button to return to Windows 95

TO RUN PROGRAM FROM DISK

Click on **Start** in Windows 95
Click on **Programs** in Start menu
Click on **MS-DOS Prompt** in Programs menu
Type **A:INF111** and then press the <Enter> button

APPENDIX G

Student Instructions – Version 2

INF111 INFORMATION SYSTEMS THEORY
Instructions for Review Program

The review program disk contains a testing program of about 170 questions. The program should run on any PC compatible computer. This test bank includes all of the questions in the common final. You are encouraged to use this program to study for the test. The instructions for running the program are as follows:

(1) Press Caps Lock button before running program.

(2) When asked to enter your name, enter both first and last name. You may enter more than one name but NO commas.

(3) First, try to run program from disk. If you receive an "unauthorized" or "invalid" message during any of the following steps, contact your instructor.

Click on **Start** in Windows 95
Click on **Programs** in Start menu
Click on **MS-DOS Prompt** in Programs menu
Type **A:** then press the <Enter> button
Type **GWBASIC INF111** and then press the <Enter> button again

(4) If step (3) above does not work, you may run the program in a Baker College computer room as follows:

Click on **Start** in Windows 95
Click on **Programs** in Start menu
Click on **MS-DOS Prompt** in Programs menu
Type **F:** then press the <Enter> button
Type **CD G:\ASSIGN\CAI** and then press the <Enter> button
Type **GWBASIC INF111** and then press the <Enter> button

(5) If you receive a message that says **Device Fault in 940** try typing **RUN** and pressing the <Enter> button to restart program. If that doesn't restart the program try typing **LOAD "INF111** and press the <Enter> button to reload the program then type **RUN** and press the <Enter> button to restart the program.

(6) After exiting the program, if the computer does not return to Windows 95, Type **EXIT** and press the <Enter> button.

(7) Don't forget to click on **Start**, then **Shut Down** before turning off your computer.

APPENDIX H

Student Instructions – Version 3

INF111 INFORMATION SYSTEMS THEORY
Instructions for Review Program

The review program disk contains a testing program of about 170 questions. The program should run on any PC compatible computer. The questions in this test bank were chosen to help you prepare for the common final. You are encouraged to use this program to study for the test. The instructions for running the program are as follows:

(1) Turn computer on. You may have to turn the monitor on separately.

(2) In the Baker College System a password dialog box will appear. Type the password **student** and press the Enter button. Note that the word student wil not appear on the screen so type it carefully.

(3) Insert your disk in the disk drive. The edge with the metal plate should go in first. The side with the metal hub should face towards the rectangular button next to the slot (usually down).

(4) To run the program:

Using the mouse, point arrow at **Start** button in lower left corner of Windows 95 screen and click left mouse button.
Using the mouse, point arrow at the **Programs** entry in the Start menu.
Using mouse, slide arrow to right, point arrow at **MS-DOS Prompt** entry in Programs menu and click left mouse button.
If the window does not fill the screen, point the arrow at the Maximize button in the upper right corner of the window → and click the left mouse button.

At the black DOS screen, type **A:** then press the <Enter> button
Then type **GWBASIC INF111** and then press the <Enter> button again

(5) When asked to enter your name, enter both first and last name. You may enter more than one name but NO commas.

(6) Errors and Problems:

If the program will not let you go on to the next chapter's questions, quit the program and restart it.
If the program seems to "lock up", hold down the Ctrl key and press the Pause - Break key
If the last line says **OK**
Type **RUN** and then press the <Enter> button
If the menu doesn't appear and the last line says **OK**
Type **SYSTEM** and press the <Enter> button,
If the last line says **A:>**
Type **GWBASIC INF111** and press the <Enter> button
If you have difficulty with this program, contact Jim Newtown at (517) 723-5251. His office hours are M&W, 12-2; Tu&Th, 10-12 and 2-4.

(7) After exiting the program, if the computer does not return to Windows 95, Type **EXIT** and press the <Enter> button.

(8) Before turning off your computer, don't forget to use the mouse to left click on **Start**, **Shut Down**, then **OK.**

APPENDIX I

Pre-Instructor Interview – Course – Version 1

1. What do you think should be the purpose of the INF111 Information Systems Theory course?
2. What do you think should be the content of INF111?
3. In what ways can we give this content to students so that they can pass the common final for INF111?
4. How much lecture time do you usually spend in this class? How much do you think is really useful?
5. Do you think a "drill and practice" computer program using a test bank could help the students pass the final?
6. Do you think such a computer program could reduce the amount of lecture time? How much?
7. After introduction, should such a program be used during class time? Individually? Available on separate disks?

APPENDIX J

Pre-Instructor Interview – Computer Instructional Aid – Version 1

1. What choices should be available to the student from the main menu? Choice of chapter? Multiple chapters? Cumulative (all chapters up to and including current one)? Complete (entire textbook)?
2. How should questions be presented? Multiple choice format? Random order? Random choices (A to D)? Repeat questions (always, only when incorrect)? Feedback?
3. What results should be provided? What printouts? What should be recorded (score, time)?
4. What should the instructor require of the student? How many and how often should printouts be required? What score (%) should be required of the student? Should the program require a specific score before allowing the student to continue to the next chapter?

APPENDIX K

Post-Instructor Interview –Version 1

1. Did you like using the computerized questions and answers?
2. Do you believe the computer program was a positive or a negative part of the course?
3. Any suggestions to make the program better?
4. Is there another review method that you think is better than the questions of the computer?
5. What problems did you encounter with the program?
6. What features of the program did you like?
7. Would you like to use the program the next time you teach this course?

APPENDIX L

Pre-Instructor Interview –Version 2

1. Do you believe this is an important class? Why or why not?
2. What do you feel should be the most important part of this class?
3. Are you satisfied that the common final appropriately tests what is important in this class?
4. What aids and tools do you use to help students pass the final exam?
5. Do you think a computer program that presents questions from a test bank might help your students review and prepare for the final?
6. How difficult is it to use a computer room during class time?
7. May I use some class time to distribute this program and demonstrate it to your students?
8. Would it be possible to require your students to turn in the printout indicating they used the computer program?
9. What questions would you like in the test bank used by the computer program?
10. Do you think the student should be able to print out the questions?
11. What results, if any, would you like the program to print out when the student ends the program?
12. Is there anything else you would like the program to do?
13. What problems and concerns do you have regarding the use of this computer program in your class?

APPENDIX M

Post-Instructor Interview –Version 2

1. Did you like using the computerized questions and answers?
2. Why do you feel this was or was not a helpful tool for this course?
3. What problems did you encounter with the program?
4. What features of the program did you like?
5. What features of the program did you NOT like?
6. Any suggestions to make the program better?
7. Did you give students credit for using this program?
8. Would you like to use the program the next time you teach this course?
9. Do you think a similar tool would be helpful for other courses?

APPENDIX N

Sample Student Printout

INF111 REVIEW PROGRAM RESULTS

JIM	5	COMPLETED	07-19-1998
	6	9	41475
JIM	5		07-19-1998
	3	4	49812

APPENDIX O

Student Survey Form

STUDENT NAME:

_____

CAMPUS : _____

DAY(S): _____ HOUR: _____

1. How useful was the software?

Very Useful	Somewhat Useful	A Little Bit	Not At All	Not Applicable
(1)	(2)	(3)	(4)	(NA)

SUGGESTIONS FOR MAKING THE PROGRAM MORE HELPFUL:

2. How easy was the software to use?

Very Easy	Easy	Somewhat Difficult	Very Difficult	Not Applicable
(1)	(2)	(3)	(4)	(NA)

PLEASE DESCRIBE ANY PROBLEMS YOU HAD:

3. Why did you use the program? What influenced you?

4. Did you receive adequate instruction in how to use the program?

Very Good	Good	Adequate	Inadequate	Not Applicable
(1)	(2)	(3)	(4)	(NA)

COMMENTS:

5. Where did you use the program? Check all that apply.

___ home
___ at a friend or relative
___ work
___ open computer lab at Baker College
___ LSS
___ other Please describe:

6. The software covered 7 chapters. Please check all chapters you used:

___ 1. An Overview of Computer Concepts
___ 2. Computer Software Applications: User Tools
___ 3. Input to the Computer
___ 4. The System Unit
___ 5. Output from the Computer
___ 6. Secondary Storage
___ 7. Communications and Networks

7. Think back over all of your use of the software. How many hours total did you use the software?

COMMENTS:

8. Please share with me any suggestions, criticisms, or advice that would help me improve the software.

Thank you for your comments and suggestions. If you are also willing to be interviewed after the end of this course, please print your name and phone number below. The interview will ask you questions about your experiences using the review program and will take about 10 to 30 minutes.

Name: _____

Phone: _____

APPENDIX P

Portion of Common Final

Directions: Select the correct answer and darken in the corresponding letter on your Scantron.

---

1. An electronic device, operating under the control of instructions stored in its own memory unit, that can accept data (input), process data arithmetically and logically, produce output from the processing, and store the results for future use is a(n) - _____.
   A. Computer
   B. Calculator
   C. Adding machine
   D. Main memory

2. The speed of laser printers is rated in _____.
   A. Lines per minute (lpm)
   B. Pages per minute (ppm)
   C. Characters per second (cps)
   D. Dots per second (dps)

3. _____ printers use a process similar to a copying machine, directing an intense light beam at a photosensitive drum to create the text or image that will be transferred to the paper.
   A. Ink-jet
   B. Dot matrix
   C. Laser
   D. Thermal

4. Printer resolution is measured by the number of _____ that can be printed.
   A. Lines per page (lpp)
   B. Characters per inch (cpi)
   C. Images per page (ipp)
   D. Dots per inch (dpi)

5. An advantage of impact printers is that _____.
   A. They are quiet and produce high-quality output
   B. Using specially treated paper, multiple copies can be created at one time
   C. They have become the standard for business correspondence
   D. All of the above

6. The monitor is _____.
   A. The visual output device of a computer
   B. An output device used to produce high-quality line drawings
   C. An output technique that records output as microscopic images on roll or sheet film
   D. Spoken words that are conveyed to the user from the computer

7. The most widely used monitors are equivalent in size to a _____ television screen.
   A. 7- to 10-inch
   B. 14- to 17-inch
   C. 21- to 24-inch
   D. 28- to 31-inch

8. The on-screen dots that can be illuminated are referred to as _____.
   A. Monitor
   B. Bits
   C. Pixels
   D. Icons

9. The speed at which the entire screen is redrawn is called the _____ rate.
   A. Resolution
   B. Refresh
   C. Pixel
   D. Diffusion

10. _____ use liquid crystal display (LCD) technology and are designed to be placed on top of an overhead projector to display an image on a computer screen to an entire group.
   A. Plotters
   B. Video display terminals
   C. Projection panels
   D. Addressable displays

11. What is soft copy?
   A. Output that is displayed on a screen
   B. Output that is printed on paper
   C. Output that has been saved on a floppy disk
   D. Output that simulates a three-dimensional environment

12. Into what two groups are printers generally classified?
    A. Tractor feed or friction feed, based on how papers is transported through the printer
    B. Impact or nonimpact, based on how characters are transferred to the paper
    C. Unidirectional or bidirectional, based on how characters are transferred to the paper
    D. Draft quality or letter quality, based on the appearance of the printed output

13. Because the diskettes introduced by IBM in the early 1970s were thin and flexible, they were often called _____, a term that is still used.
    A. Ductile disks or duckies
    B. Skinny disks or skinnies
    C. Floppy disks or floppies
    D. Flimsy disks or flimsies

14. A _____ is a narrow recording band forming a full circle around a diskette.
    A. Track
    B. Cylinder
    C. Partition
    D. Cluster

15. A sector is _____.
    A. An area that can function as if it where a separate disk
    B. All tracks of the same number
    C. A pie-shaped section of a diskette
    D. The smallest unit of diskette space used to store data

16. A billion bytes of information is called a _____.
    A. Terabyte
    B. Megabyte
    C. Nanabyte
    D. Gigabyte

17. Where are programs and data stored when they are **not** being processed?
    A. Main memory
    B. Primary storage
    C. RAM (random access memory)
    D. Auxiliary storage

18. _____, which can be considered verbal electronic mail, digitizes a caller's message so it can be stored on a disk like other computer data.
    A. Voice mail
    B. Teleconferencing
    C. FAX
    D. Telecommuting

REFERENCES

REFERENCES

Altimirano, L. C. (1998). A teacher's dual roles as a client and change agent: An examination of the adoption of and teaching with hypermedia technology. Unpublished doctoral dissertation, Michigan State University.

Anonymous (2000). Retrieved July, 10, 2000 from The People's Cyber Nation on the World Wide Web: http://www.cyber-nation.com/victory/quotations/subjects/quotes_memory.html

Aronson, D. T., & Briggs, L. J. (1983). Contributions of Gagne and Briggs to a prescriptive model of instruction. In C. M. Reigeluth (Ed.), Instructional-design theories and models: An overview of their current status (pp. 75-100). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Bergman, R., & Moore, T. (1990). Managing interactive video/multimedia projects. Englewood Cliffs, NJ: Educational Technology.

Bloom, B. S., (Ed.). (1956). Taxonomy of educational objectives, Handbook I: Cognitive domain. New York: David McKay.

Branson, R. K. et al. (1975). Interservice procedures for instructional systems development.
Washington: Government Printing Office.

Brethower, D., & Smalley, K. (1998). Performance-based instruction: Linking training to business results. San Francisco: Jossey-Bass Pfeiffer.

Cavalier, J.C., and Klein, J.D. (1998). Effects of cooperative versus individual learning and orienting activities during computer-based instruction. Educational Technology Research and Development, 46, 1, 5-17.

Chambers, J. A. & Sprecher, J. W. (1984). Computer-assisted instruction: Current trends and critical issues. In D. F. Walker & R. D. Hess (Eds.), Instructional Software: Principles and perspectives for design and use (pp. 6-20). Belmont, CA: Wadsworth.

Clark. R. R. (1985). Confounding in educational computing research. Journal of Educational Computing Research, 1, 137-148.

Davis, R. H., Alexander, L. T. and Yelon, S. L. (1974). Learning system design: An approach to the improvement of instruction. New York: McGraw-Hill.

Diamond, R. M. (1989). Designing & improving courses and curricula in higher education: A systematic approach. San Francisco: Jossey-Bass.

Dick, W., & Carey, L. (1990).). The systematic design of instruction (3rd ed.). Glenview, IL: Scott, Foresman.

Dick, W., & Carey, L. (1996). The systematic design of instruction (4th ed.). Glenview, IL: Scott, Foresman.

Firdyiwek, Y. (1999). Web-based courseware tools: Where is the pedagogy? Educational Technology, 39, No 2, 29-34.

Fleming, M. L. (1987). Displays and communication. In R.M. Gagne (Ed.), Instructional technology: Foundations (pp. 233-260). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Gagne, R. M. (1977). The Conditions of Learning (3rd ed.). New York: Holt, Rinehart & Winston.

Gagne, R. M., & Briggs, L. (1974). Principles of instructional design. New York: Holt, Rinehart, Winston.

Gagne, R. M., Briggs, L. J., & Wager, W. W. (1992). Principles of instructional design (4th ed.). New York: Holt, Rinehart & Winston.

Gagne, R.M., Wager, W., & Rojas, A. (1984). Planning and authoring computer-assisted instruction lessons. In D. F. Walker & R. D. Hess (Eds.), Instructional Software: Principles and perspectives for design and use (pp. 57-67). Belmont, CA: Wadsworth Publishing Company.

Gayeski, D. M. (1999). Multimedia learning systems: Technology. In H. D. Stolovitch & E. J. Keeps (Eds.), Handbook of Human Performance Technology (2nd ed., pp. 589-605). San Francisco: Jossey-Bass Pfeiffer.

Gropper, G. L. (1983). A behavioral approach to instructional prescription. In C. M. Reigeluth (Ed.), Instructional-design theories and models: An overview of their current status (pp. 101-161). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Gustafson, K. L. (1991). Survey of Instructional Development Models. Syracuse, New York: Information Resources Publications, Syracuse University.

Gustafson, K. L., & Tillman, M. H. (1991). Introduction. In L. J. Briggs, K. L. Gustafson & M. H. Tillman (Eds.), Instructional design: Principles and applications (2nd ed.). Englewood Cliffs, NJ: Educational Technology.

Havelock, R. G. (1973). The change agent's guide to innovation in education. Englewood Cliffs, NJ: Educational Technology.

Havelock, R.G. (1995). The change agent's guide to innovation in education (2nd ed.). Englewood Cliffs, NJ: Educational Technology.

Hentrel, B. K., & Harper, L. (1985). Computers in education: A guide for educators. Ann Arbor: University of Michigan Press.

Jamison, D., Suppes, P., & Wells, S. (1974). The effectiveness of alternative instructional media: A survey, Review of Educational Research, 44, 1-68.

Kaufman, R. A. (1972). Educational System Planning. Englewood cliffs, NJ: Prentice-Hall.

Kearsley, G. (1998). Educational technology: a critique. Educational Technology, 38, 2, 47-51.

Keller, A. (1987). When machines teach: Designing computer courseware. New York: Harper & Row, Publishers.

Kemp. J. (1977). Instructional design. Belmont, CA: Fearon-Pitman.

Lawler, C., Rossett, A., and Hofman, R. (1998). Using supportive planning software to help teachers integrate technology into teaching. Educational Technology, 38, 5, 29-34.

Leh, A.S.C., Sleezer, C.M., and Anderson, V.A. (1998). Measuring the value of educational technology in different contexts. Educational Technology, 38, 4, 28-33.

Leshin, C., Pollock J., & Reigeluth, C. (1990) Instructional design: Strategies & tactic for improving learning and performance. Englewood Cliffs, NJ: Educational Technology.

Mager, R. F. (1997). Making instruction work or skillbloomers: A step-by-step guide to designing and developing instruction that works (2nd ed.). Atlanta: Center for Effective Performance.

Merrill, M. D. (1983). Component display theory. In C. M. Reigeluth (Ed.), Instructional-design theories and models: An overview of their current status (pp. 279-333). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Moallem, M., & Earle, R. S. (1998). Instructional design models and teacher thinking: Toward a new conceptual model for research and development. Educational Technology, 38, 2, 5-22.

Noble, D. (1998). Digital diploma mills: the automation of higher education. Summarized by the editors of Educom Review. Educom Review, 33, No 3, 22-25.

Nugent, G. C. (1987). Innovations in telecommunications. In R.M. Gagne (Ed.), Instructional technology: Foundations (pp. 261-282). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Office of Technology Assessment (1988). Power on! New tools for teaching and learning. Washington: Government Printing Office.

Perelman, L. J. (1987). Technology and transformation of schools. Alexandria, VA: National School Boards Association.

Provenzo, E. F., Jr. (1986). Beyond the Guttenberg galaxy: Microcomputers and the emergence of post-typographic culture. New York: Teachers College Press.

Resta, P. E. (1984). The improvement of instruction through technology: Promises for the future. In National conference on communications technology in education and training. Silver Spring, Maryland: Information Dynamics.

Rickard, W. (1999). Technology, higher education, and the changing nature of resistance. Educom Review, 34, No 1, 42-45.

Rizza, P. J., Jr. (1981). How can we best produce courseware? (The process of courseware development). In Technology and education: Policy, implementation, evaluation (pp. 275-289). Washington: Institute for Educational Leadership.

Rogers E. M. (1995). Diffusion of innovations. New York: Free Press

Romiszowski, A. J. (1981). Designing instructional systems. London: Kogan-Page.

Rosenberg, M. J., Coscarelli, W. C., & Hutchison, C. S. (1999). The origins and evolution of the field. In H. D. Stolovitch & E. J. Keeps (Eds.), Handbook of Human Performance Technology (2nd ed., pp. 24-46). San Francisco: Jossey-Bass Pfeiffer.

Rowntree, D. (1982). Educational technology and curriculum development. London: Harper & Row.

Scanlon, E., Tosunoglu, C., Jones A., Butcher, P., Ross, S., Greenberg, J. Taylor, J. and Murphy, P. (1998). Computers & Education, 30, 1/2, 9-14.

Seels, B., & Glasgow, Z. (1990). Exercises in instructional design. Columbus: Merrill.

Shavelson, R. J. (1988). Statistical reasoning for the behavioral sciences, (2nd ed.). Boston: Allyn and Bacon.

Slawson, D., Shaughnessy, A., & Bennett, J. (1994). Becoming a medical information master: Feeling good about not knowing everything. The Journal of Family Practice, 38, 505-513.

Strudler, N. & Gall, M. (1988, April). Successful change agent strategies for overcoming impediments to microcomputer implementation in the classroom. Paper presented at the annual meeting of the American Educational Research Association, New Orleans, LA. (ERIC Document Reproduction Service No. ED 298 938).

Taylor, F. (1912). Scientific management. New York: Harpers.

Tannenbaum, R. (1999). Education or training: Reflections of a life in computing. Educom Review, 34, No 1, 10-15.

Twelker, P.A. et al. (1972). The systematic development of instruction: An overview and basic guide to the literature. Stanford, CA: Stanford University, ERIC Clearinghouse on Educational Media and Technology.

Van Patten, J. (1989). What is instructional design? In K. Johnson & L. Foa (Eds.). Instructional design: New alternatives for effective education and training. New York: Macmillan.

Weizenbaum, J. (1976). Computer power and human reason: From judgment to calculation. New York: W. H. Freeman.

Wilson, B., (1999). Adoption of learning technologies: Toward new frameworks for understanding the link between design and use. Educational Technology, 34, 1, 12-16.

Wilson, B., Dobrovolny, J., & Lowry, M. (1999). A critique of how technology adoption models are utilized. Performance Improvement, 38 (5), 24-29.

Yelon, S. L. (1996). Powerful principles of instruction. White Plains, New York: Longman,

Yelon, S. L., & Sheppard, L. M. (1999). The cost-benefit transfer model: An adaptation from medicine. Performance Improvement Quarterly, 12 (3), 78-93.

Systems Development Life Cycle Textbooks

Burch, J.G. (1992). Systems Analysis, Design, and Implementation. Boston: Boyd & Fraser Publishing Company.

Capron, H.L. (1986), Systems Analysis and Design. Reading, MA: The Benjamin/Cummings Publishing Company, Inc.

Cats-Baril, W.L., and Thompson, R.L. (1997). Information Technology and Management. Chicago: Irwin.

Edwards, P. (1985). Systems Analysis, Design, and Development with Structured Concepts. New York: Holt, Rinehart and Winston,

Gore, M., & Stubbe, J.W. (1988). Elements of Systems Analysis (4th ed.). Dubuque, IA: Wm. C. Brown Publishers.

Laudon, K.C., and Laudon, J.P. (1997). Essentials of Management Information Systems Organization and Technology (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

Licker, P.S. (1987). Fundamentals of Systems Analysis with Application Design. Boston: Boyd & Fraser Publishing Company.

Oz, E. (1998). Management Information Systems. Cambridge, MA: Course Technology.

Powers, M.J., Cheney, P.H., & Crow, G. (1990). Structured Systems Development: Analysis, Design, Implementation (2nd ed.). Boston: Boyd & Fraser Publishing Company.

Shaw, J.C. and Atkins, W. (1970). Managing Computer System Projects. New York: McGraw-Hill Book Company.

Shelly, G.B., Cashman, T.J., & Rosenblatt, H.J. (1998). Systems Analysis and Design (3rd ed.). Cambridge, MA: Course Technology.

Taylor, F. (1912). *Scientific Management.* New York: Harper's.

Whitten, J.L., Bentley, L.D., and Barlow, V.M. (1994). Systems Analysis and Design Methods (3rd ed.). Burr Ridge, IL: Irwin.

Wohl, A.D., and Hunt, C.S. (1991). Managing Integrated Business Systems: A Case Approach. Cincinnati, OH: South-Western Publishing Co.